Subharmonic and Non-Subharmonic Pulsed Control of Thermoacoustic Instabilities: Analysis and Experiment

by
J. Matthew Carson

Thesis submitted to the Faculty of the
Virginia Polytechnic and State University
as partial fulfillment of the requirements for the degree of

Master of Science
in
Mechanical Engineering

Approved:

_____
William R. Saunders

_____ _____
William T. Baumann Donald J. Leo

December 13, 2001

Blacksburg, Virginia

Keywords: Thermo-acoustic Instabilities, Active Combustion Control, Pulsed Control, Subharmonic Control, Linear Phase Shifter

Subharmonic and Non-Subharmonic Pulsed Control of Thermoacoustic Instabilities: Analysis and Experiment

J. Matthew Carson
Virginia Polytechnic Institute and State University
Advisor: Dr. William R. Saunders, Chair

Abstract

Thermoacoustic instabilities are a problem in modern pre-mixed combustors causing reduced performance and leading in the extreme to combustor failure from excessive pressure cycles. Much work has been done using linear controllers to eliminate these instabilities. Many experimenters in the field have used pulsed and subharmonic fuel controllers to eliminate these instabilities, but very little investigative work has been done on these controllers. The goal of this work is to explain the mechanism of control behind pulsed controllers. It is shown that the combustion system can be treated as a linear system, thus meaning that frequency components of the control signal at the desired instability frequency are the dominant means of control, with nonlinear effects only serving to slightly reduce the gain necessary for control. Fourier analysis is thus performed on pulsed signals and the components analyzed, showing that there will indeed be a component of a pulsed signal at the frequency of the instability, aside from a few select cases. It is then proven that this frequency component is largely responsible for control of the thermoacoustic system using proportional height pulse train signals, which will change pulse height based on the amplitude of the instability. This analysis is then used to predict the height of instabilities resulting from the use of fixed height pulse control signals. Finally, numerical simulations and experimental observations support the analytical constructs. Acoustic control is mainly used for these experiments, although some preliminary work with liquid fuel controllers is also presented.

Acknowledgements

There have been many people who have helped make this research possible. I would like to thank Virginia Tech for accepting me for my graduate studies and for providing me with the support to be able to return to school. I also thank the Office of Naval Research for providing the grant that financed the research in this thesis.

I owe my advisor, Dr. Will Saunders, a debt of gratitude. I have learned a great deal during my time here at Virginia Tech from Dr. Saunders. He has a positive outlook on life and on his students and I appreciate the latitude and trust he gave me to complete my research in my own way to enhance my learning. I would also like to thank Dr. William Baumann for all the help he provided in understanding the mathematics and control concepts presented in this thesis, and for the tremendous help given with experimental processes. I would also like to thank the other professors from the Virginia Active Combustion Control Group, Dr. Vandsburger and Dr. West, for their support of the students in the group who have directly and indirectly helped with my experiments.

I would like to thank the other students in the group, particularly Mike Vaudrey for his help in learning how to program on the DSpace controller. I also owe thanks to Ernie, Ludwig, John, Steve, and Wajid for their help with the liquid fueled reactor. Also thanks to Ernie, Colin, Salahi, John, and Aaron for being great office mates who made my time on campus fun.

I also would like to thank my roommates Jason and Mark for putting up with me for the last year and a half. We never had a cross word and they had a lot of fun distracting me from my thesis work. I also thank my other friends from the Baptist Student Union for all the help and support given, and for all the fun times we had while I was a student here both in undergrad and grad school. You will now have to find a new old person for the BSU! I thank my friend Rob Henry for always being there when I needed to talk. I always enjoy taking trips with you and I look forward to many more ski trips when I get vacation! Thanks also to Rob Kilgore and Keith Sherrill for your continuing friendship. It is nice to have lifelong friends.

Last but certainly not least, I would like to thank my family. I thank my sisters, Laura and Cathy, and their husbands, Clay and Tim, for their support. I have grown closer to you the older I have gotten, and I enjoy spending time with all of you. I have been especially blessed to be able to get to know Clay and Tim, and I am proud to have them for brothers-in-law. My parents, Jim and Linda, have always supported me in everything I have endeavored to do. I thank you for your support while I was in grad school, as it was an adjustment to return to school. I thank you for raising me right and for providing a strong moral background and a good home, and I hope to provide the same kind of household for my family in the future.

Matthew Carson

Blacksburg, VA
December 2001

Table of Contents

Table of Figures

Table of Tables

Nomenclature

A	Amplitude of instability
c_k	k^{th} Fourier coefficient
c_M	Fourier coefficient at instability frequency
c_n	n^{th} Fourier coefficient
d	Duty cycle
DF	Describing function
f_0	Frequency of instability (Hz)
k	Gain of controller
k_{ult}	Controller gain at which secondary instabilities are generated
M	Subharmonic ratio
$p'(t)$	Unsteady acoustic pressure
$q'(t)$	Unsteady heat release
R	Rayleigh index
t	Time
T, T_0	Period of instability (s)
w	Pulse width (s)
X	Pulse height (V)
φ	Equivalence ratio
$\varphi(A,\omega)$	Phase shift of controller (rad)
τ	User configurable time delay of controller (s)
ω	Frequency of instability (rad/s)

1 Introduction

The last decade has seen an increasing level of environmental awareness. With this awareness have come new laws regarding emissions that have impacted continuous combustion processes. Increased performance is also being demanded to meet today's energy needs and to propel newer, faster aircraft. In addition, better fuel economy is desired for cheaper energy in the case of industrial gas turbines and longer range in the case of jet aircraft. Non-optimal combustion temperatures and larger air/fuel ratios are used to achieve the sometimes-contradictory goals of lower emissions, increased performance, and improved fuel economy at the expense of combustion stability. These increases in efficiency push the envelope of design, and require control systems that were unnecessary in the past. Just as automatic flight control systems in modern fighter jets provide for unheard of maneuverability by enabling an unstable airplane to be flown, modern automatic control systems can make an unstable combustion process meet unheard of performance and environmental goals.

Control systems are necessary because the new combustor operating conditions can in general cause thermoacoustic instabilities. These instabilities can be seen as an undamped pressure oscillation in the combustor. These oscillations, in extreme cases, can cause significant damage due to extremely high pressures in the combustor, and will reduce component life due to the increased cycling of the combustor parts. In any case, these unstable oscillations will have to be accounted for and may cause heavier, more expensive components to be used to protect against the instability. In addition, the instabilities can cause a reduction in the performance of the combustor. Some of the heat generated by the combustor will contribute to the unstable pressure oscillation and will thus not be available for power generation. In some cases, the large pressure fluctuation may even cause the flame to be extinguished.

One of the goals of the Virginia Active Combustion Control Group has been to study advanced control methods and to examine their usefulness in eliminating these combustion instabilities. Other institutions have done work on control systems, but most

of the people studying this problem have been looking at the problem from a combustion engineer's standpoint. The goal of this group is to allow control system engineers to examine the problem and develop innovative ways to control combustion systems.

First it is necessary to examine what has been done in the past and ensure that it can be explained from a controls perspective. This will provide some encouragement that the mechanisms driving the instabilities are understood correctly. This thesis in particular will examine pulsed and subharmonic controllers, and explain why each works. It will show that the combustion system can be thought of primarily as a linear system, and the widely-used linear phase shift controller will be used as a baseline for examining how a pulsed controller works. It will conclude that the mechanisms behind the different types of controllers are identical, and that the results from a limit cycle amplitude curve gained from a linear phase shifter can be used to design pulsed fuel controllers to meet certain specifications.

To examine this method of control, pulsed control was applied to a lean pre-mixed methane burning Rijke tube. This tube generates well-defined instabilities and has been the basis for examining linear phase shift controllers in the past. Pulsed control is the desired form of control because it allows off-the-shelf solenoid fuel injectors to be used to implement control. Loudspeakers have in general been used to control systems proportionally, but these loudspeakers do not have the robustness or control authority to eliminate instabilities in industrial scale combustors.

1.1 Thermoacoustic Instabilities

Before instabilities can be eliminated, they must be understood. This section will provide a brief overview of the instabilities in lean pre-mixed combustors, known as "thermoacoustic instabilities." The mechanism driving these instabilities has been known for over one hundred years, and was described by Lord Rayleigh in 1878. The Rayleigh index describing the growth rate of the instability is given by

$$R = \int_0^T P'(t)q'(t)dt$$

where T is the period of the instability, $P'(t)$ is the unsteady acoustic pressure, , and $q'(t)$ is the unsteady heat release of the combustor. The index R describes the growth rate of pressure oscillations in the system. Thus a positive value of R will indicate that the unsteady heat release will be amplified by the acoustic characteristics, resulting in an instability in the pressure of the system. A negative value of R will indicate that the acoustics of the system attenuate the unsteady component of heat release, thus allowing the system to stabilize. The goal of a control system is thus to lower the Rayleigh index to a negative value.

The physical realization of the self-excited system shown by Lord Rayleigh is shown in Figure 1-1, and shows how the acoustics and the heat release feed each other to produce an unsteady pressure. A controller in this case is inserted to attempt to influence the acoustic pressure of the system to eliminate the instabilities with the use of a loudspeaker. For a fuel control system in general the controller will feed into the heat release.

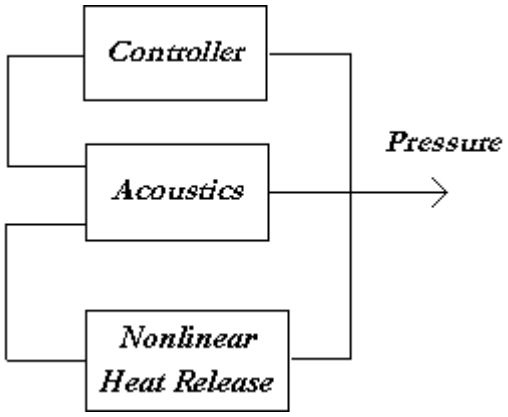


Figure 1-1: Combustion self-excited loop

A positive Rayleigh index indicates a positive growth rate of instability. A linear system would thus allow the instability to grow unbounded. However, in any physically realizable system, this is not possible. In reality, there exist physical limitations on the pressure that can be generated in the combustor. Thus the system can be thought of as linear until a nonlinear limiting factor arises. This limiting factor will cause the system to stabilize at the maximum combustor pressure that can be physically realized. For this reason, the instability is known as a "limit cycle." It will be shown that in general these nonlinearities can be ignored when designing a control system. Once the system is

linearly stable, the pressure oscillations will be below the nonlinear limit and will thus not be affected by the nonlinearities.

1.2 Control Methods

Many different types of passive and active controllers have been examined for use in combustion processes. Passive methods are in general not as effective, as they only compensate for a particular instability and are not able to eliminate other instabilities as they arise. A problem with passive suppression is that the combustion plant must be well understood to eliminate the instability. Devices such as resonators can be used, but these are only effective at one particular frequency. Any frequency shift will cause the system to go unstable. Instabilities may also be eliminated via combustion chamber design to eliminate resonances. However, this is a difficult proposition because the acoustic properties depend on the physical dimensions of the combustor. Thus small-scale models as used for development work are not able to predict instabilities in full-scale combustors, and the process becomes one of trial and error. In addition, the design of a combustor to eliminate one instability may inadvertently create another.

The limitations of passive control strategies have encouraged people to pursue active combustion control strategies. These are discussed in detail in Section 2.2. Active control schemes have the advantage of being able to adapt to the instability and in general will have better attenuation than passive strategies. Some active controllers include linear phase shifters, LQG-LTR, H_∞, neural network, pulsed, and subharmonic pulsed. All have advantages and disadvantages. This thesis is primarily concerned with pulsed and subharmonic pulsed, although baseline work was done with a linear phase shifter.

1.3 Thesis Overview

The main goal of this thesis is to provide an explanation of pulsed control of thermoacoustic instabilities. To accomplish this, experiments were done on a laboratory scale lean pre-mixed tube combustor known as a Rijke tube. The Virginia Active Combustion Control Group has used this tube for a while, and its characteristics are well known (see Nord 2000 [11]). As such, it provides a good baseline for examining control.

The tube has a well-developed instability at approximately 178 Hz, and it has been controlled with a linear phase shifter in the past.

After this brief introduction, relevant literature will be discussed in Chapter 2. This will show that, while there has been a good deal of work done with pulsed control, it has not been examined from a mathematical or controls viewpoint. It will be seen that this work is a natural extension of, and directly applicable to, the work done by other investigators. It will also show that many of the results seen by other investigators may be explained by the control theory presented in Chapter 3.

Chapter 3 provides a mathematical overview of pulses. It breaks down the frequency components of pulse train signals (subharmonic and non-subharmonic) and examines the gain of corresponding controllers using describing functions. It shows how the duty cycle of the pulse is integral to the gain of the controller through a nonlinear relationship, and explains how using different subharmonic ratios and varying duty cycles can modify the gain to add efficiency and maintain control.

Chapter 4 begins by giving the results of a simulation run to examine pulsed control systems, and shows that only the component of the pulse train signal at the instability frequency is responsible for control of the system. It then shows how this theory was verified experimentally. This was done first by using a proportional height, varying duty cycle pulse train signal to find which component was implemented in control. This information was then used to predict the heights of instabilities remaining when fixed pulse height signals were used.

Chapter 5 provides a summary of results and draws conclusions about pulsed control from them. It also provides an overview of possible advantages that can be achieved with pulsed control through further research, and gives some ideas for continuing research in this field.

Appendix A provides a detailed description of the C code used to perform the control on the system, and contains the code itself.

Appendix B explains how this code was implemented on the DSpace controller, how the user interface was created, and how the controller is used.

2 Literature Review

This section gives an overview of relevant work done on the topic of control of thermoacoustic instabilities. It will also show why the work presented in this thesis is of particular importance and can explain much of the phenomena seen by other experimenters in this field.

2.1 Thermoacoustic Instabilities

Although a thorough understanding of the mechanisms driving combustion instabilities are not necessary to understand the theory in this thesis, a background is provided here to give some idea of where combustion instabilities come from.

Recent regulations have pushed for a reduction in the production of nitrogen oxides (NO_x) from combustion processes. Richards et al. [12] provides a thorough background of the processes involved in not only gas turbines, but also reciprocating engines and fuel cells.

Nitrogen oxides are formed when nitrogen from the combustion air combines at high temperatures with oxygen in the exhaust stream, forming either NO or NO_2. Nitrogen oxides have been known to contribute to smog and acid rain, and it is desirable to eliminate their production in combustion processes.

Many ways have been examined to rid the exhaust of combustion processes of nitrogen oxides. One method is selective catalytic reduction (SCR), which is accomplished by injecting ammonia into the exhaust stream to combine with the NO_x and break it down into atmospheric nitrogen. However, this only works for a very selective range of temperatures (560-670 K) and thus other processes must be added to set the exhaust temperature. In addition, the systems are very bulky and expensive, and thus impractical.

Another approach is to lower the temperature of combustion. Although this may contribute to other types of emissions such as hydrocarbons and carbon monoxide, it will

eliminate nitrogen oxide production because the process will not be at a high enough temperature to promote formation of these pollutants.

One way to lower the temperature of combustion is to dilute the fuel by injecting a non-reacting material such as water. However, this will not produce low enough NO_x emissions to satisfy government regulations, since high levels of water injection will drastically increase CO production and reduce flame stability before the NO_x goals are met. In addition, steam is an undesired by-product, and impurities in the water can foul turbine hardware.

Therefore engine developers have turned to lean pre-mixed combustion. This lowers the temperature of combustion by burning the fuel in the presence of excess air. Equivalence ratios approaching 0.5 have been examined, and will inject twice as much air as necessary. This is shown conceptually in Figure 2-1, where fuel and air are mixed at the inlet to the combustor.

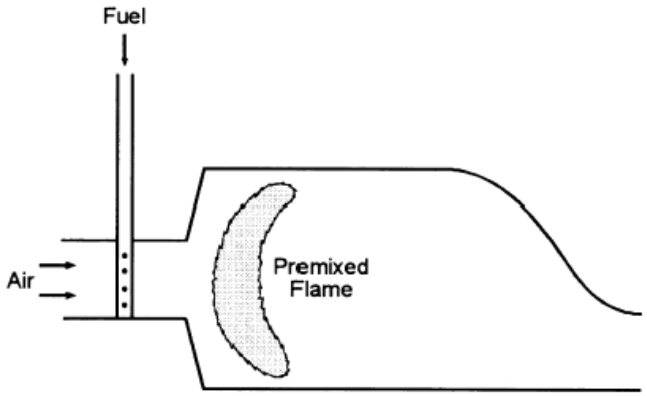


Figure 2-1: Lean pre-mixed combustion

There will be problems associated with a lean pre-mixed flame, including lean blowout and a loss of static stability. However, this thesis is concerned with dynamic stability, which will be discussed briefly here.

Air and fuel enter by the same injector in a lean pre-mixed combustor. Therefore any pressure oscillation in the combustion chamber can enter the fuel/air mixture through the fuel injector. This will then change the fuel delivery rate, and thus the heat release rate. Since this change in heat release will occur very quickly relative to the oscillation period for low frequencies, the unsteady heat release will be nearly in phase with the pressure oscillation. If this heat release oscillation occurs at a resonant frequency of the

combustion chamber, the pressure oscillations will be enhanced instead of damped. Thus the pressure oscillations will continue to grow, and cause further oscillations in the heat release. This self-excited loop thus causes the thermoacoustic instabilities.

This feedback mechanism of thermoacoustic instability generation is enhanced by Rayleigh's criterion, which states that if energy is added to the acoustic field with the right phase, unstable pressure oscillations can result. Culick [2] describes Rayleigh's criterion in detail. Rayleigh's criterion determines the phasing which must be achieved between the unsteady heat release and the acoustic wave. The criterion also states that if unsteady heat is added at a moment of greatest compression or taken away at a moment of greatest rarefaction, oscillations in the pressure field will result. If Rayleigh's criterion is satisfied and the pressure oscillation that is created is at a resonant frequency of the combustion chamber, a self-excited system will be created which will produce a thermoacoustic instability.

2.2 Control of Thermoacoustic Instabilities

There have been many types of controllers examined for use in eliminating thermoacoustic instabilities from premixed lean tube combustors. An important feature of the studied controllers is that they are all linear controllers. Below are some of the controllers that have been designed theoretically and implemented experimentally.

2.2.1 General linear control

One of the basic principles behind the ideas examined in this thesis is the idea that linear control is capable of suppressing thermoacoustic instabilities in nonlinear tube combustors. Fannin et al. [3] in 1998 examined this hypothesis, and showed that linear models are sufficient to predict combustion instabilities. However, ignoring the nonlinear effects means there will be an inability to predict the level of the instability or to accurately predict such effects as hysteresis.

Linear theory is appropriate for understanding system stability since even a nonlinear system is stable if the eigenvalues of the linearized system (about some operating point) are in the left half plane of the complex coordinate system. It is shown that the combustion system is fully controllable for any initial condition. Due to this

controllability, a linear controller is able to move the eigenvalues to the left half plane and ensure stability regardless of the nonlinearities.

Analysis of Hopf bifurcations was also done. These bifurcations can be used to predict hysteresis in a system. Hysteresis in this system is seen when the gain of the controller is increased past the gain necessary to stabilize the system and then decreased. For values of the gain in the hysteresis region, the system will not achieve the same level of stability for an increasing gain as it does for a decreasing gain. Although the hysteresis is not integral to the analysis in this thesis, it is noted, and Fannin et al. show that it can be explained.

Finally analysis was done using a linear controller and a nonlinear controller. It was shown that the linear controller could eliminate the instability, while the nonlinear controller was only able to reduce the level of it. This is a result of the fact that the linear controller is able to move the eigenvalues and the bifurcation, while the nonlinear controller can only change the shape of the bifurcation.

Thus Fannin's main conclusion, which is significant for the work done in this thesis, is that the linear regime of the combustion system is dominant, and therefore linear control is sufficient to eliminate the instability.

2.2.2 Linear phase shifter

One of the simplest controller designs is a linear phase shifter, which simply delays the pressure signal and reinserts it into the system. Normally a loudspeaker or other acoustic driver is used for this, as fuel injectors are of the on/off variety and thus are not able to track the pressure signal. Saunders et al. in 1999 [14] examined the mechanisms behind the linear phase shifter, and showed that the way a linear phase shifter is implemented has effects on the stability of the system.

One of the main problems with a linear phase shifter is the introduction of secondary instabilities, or what may be referred to as "peak-splitting." The instability is reduced, but other, previously non-existent instabilities arise on either side of the original instability frequency. Predicting these instabilities for certain cases shows that the mechanism driving these secondary instabilities is known. For example, it is theorized that the phase roll-off from sharp bandpass filters used to separate the instability

frequency from noise in the input signal is responsible for the secondary peaks. Insufficient gain or phase margin in the open loop frequency response function (FRF) of the system with the controller in place will indicate a secondary instability.

To find the open loop frequency response function, the closed loop control system was broken before the input to the bandpass filter entering the controller. A sine wave was input instead, and the response of the system at the sine wave frequency was found. By combining the response from several sine waves (a sine sweep) the open loop FRF was measured experimentally.

The combustor system contained an instability at 178 Hz. Due to the large pressure in the tube at this and very adjacent frequencies, sine waves close to the limit cycle frequency could not be used for the FRF, as the speaker could not physically deliver enough sound power to overcome the combustion noise, resulting in low coherence. Although the system is highly nonlinear, this linear method of finding the FRF is valid. The nonlinearity serves only to limit the amplitude of the instability. At frequencies other than the instability frequency, the system is dominantly linear. This is borne out by the high coherence (>0.95) that is seen using the sine sweep method.

Since it was theorized that the phase roll-off of the bandpass filters is responsible for the secondary instabilities, two different bandpass filters were examined – one from 150-200 Hz, and the other from 150-500 Hz. Sine sweeps were performed with each in place, and the open loop FRF examined. It was found that raising the controller gain, although improving control at the limit cycle frequency, reduced open loop gain margins at the frequencies where the phase passed through 0° or –360°. When the loop was then closed, these gains caused instabilities to grow at these frequencies. As the gain or phase shift of the controller was changed, the frequencies of the secondary instabilities changed. This behavior can be seen graphically in Figure 2-2, with instabilities growing when the gain is raised.

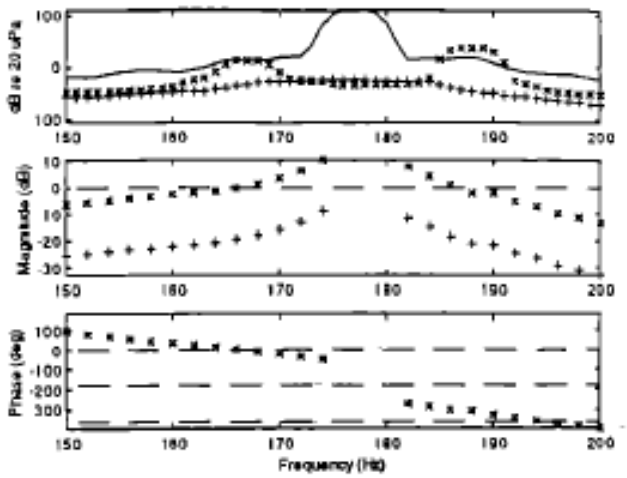


Figure 2-2: Controller induced instabilities with increase in gain (168 Hz, 189 Hz)

There are thus tradeoffs when implementing a linear phase shifter. A wider bandpass filter can move the secondary instabilities farther in frequency away from the primary instability frequency. However, there will be a reduced level of control at the instability frequency since less gain can be used due to the lower gain margin of this controller at the crossover frequency. It is important therefore in designing a phase shifter controller to examine filters and the gain used, so as to prevent secondary instabilities. These secondary instabilities will be seen and examined in this thesis.

2.2.3 Adaptive controllers

Vaudrey and Saunders [15] showed an adaptive extension of the linear phase shifter in 2000. A neural network was used to adapt the gain and the phase of an acoustic linear phase shifter to maintain control with varying equivalence ratios and flow rates. The controller uses four temperature measurements, as well as the flow rate and equivalence ratio, to calculate the open loop transfer function of the system. The neural network was trained by inputting the temperature measurements and the sine dwell transfer function data from eleven different operating conditions. To verify the ability of the neural network to learn a new condition, a twelfth operating condition was then chosen. The neural network controller was able to adjust the gain and phase of the linear phase shifter to eliminate the instability.

This adaptive controller was then compared to a standard manually configured linear phase shifter. The manual controller could not maintain stability over time as the plant operating conditions changed. In addition, the manually configured controller needed more gain to achieve stability due to a non-optimal phase setting. This greater gain for the manual phase shifter resulted in secondary instabilities being generated.

Therefore the benefits of the neural net controller are its ability to adjust to all operating conditions while maintaining a high level (~50 dB) of attenuation, and its ability to prevent secondary instabilities from forming.

Another experiment was mentioned that is worth noting. Acoustic sine waves were input into the system in the open loop with a stable flame. At high levels, the sine waves no longer appeared as a discrete frequency in the pressure spectrum, but caused a broadband increase in the pressure spectrum. The higher the level of forcing, the more bandwidth of the spectrum is raised. This does give some credibility to the idea seen in Section 4.5.1 that subharmonics of the system can have an effect on the pressure spectrum at the instability frequency.

2.2.4 Model-based controllers

An optimal controller can be designed for a thermoacoustic system if an accurate model of the system is available. Annaswamy et al. [1] attempted this for a bench-top propane combustor rig. A dynamic model of the combustor was derived based on physical parameters of the system, such as the mean flow rate and positions of the actuator and pressure transducer. This dynamic model was then used with two types of optimal controllers, LQG-LTR and H_∞.

The LQG-LTR controller has the form

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu + H(y - C\hat{x}) \\ u &= -K\hat{x} \end{aligned} \tag{1}$$

where the estimator gain is H and the state feedback gain is K. These gains are then determined from the Matlab control toolkit using the cost function

$$J = \int_0^\infty (y^T Q y + u^T R u)\,dt \qquad Q = I, \qquad R = \rho I \tag{2}$$

The weighting value ρ can be fine-tuned to provide optimal controller performance. The value of ρ=0.1 was chosen to due to a physical acceleration limit of 600 m/s^2 of the loudspeaker actuator. This controller was then implemented on the bench top rig at two different actuator and sensor positions and compared to the theoretical results. The experimental response is shown in Figure 2-3 and the theoretical response is shown in Figure 2-4 for ϕ=0.7.

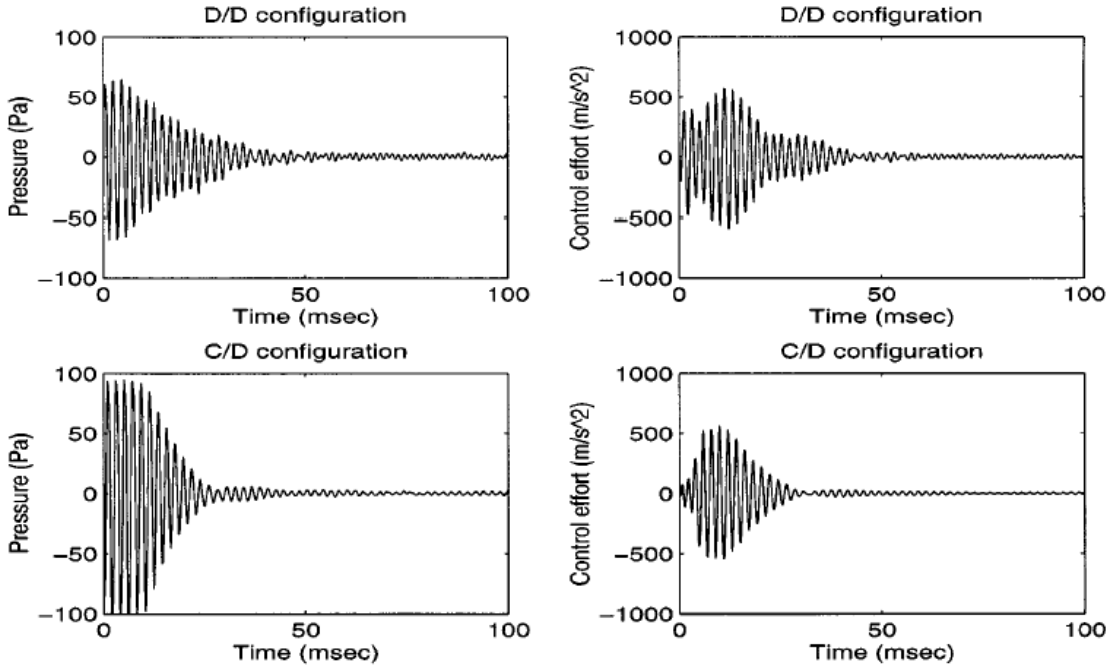


Figure 2-3: Experimental results for LQG-LTR controller at two locations, ϕ=0.7.

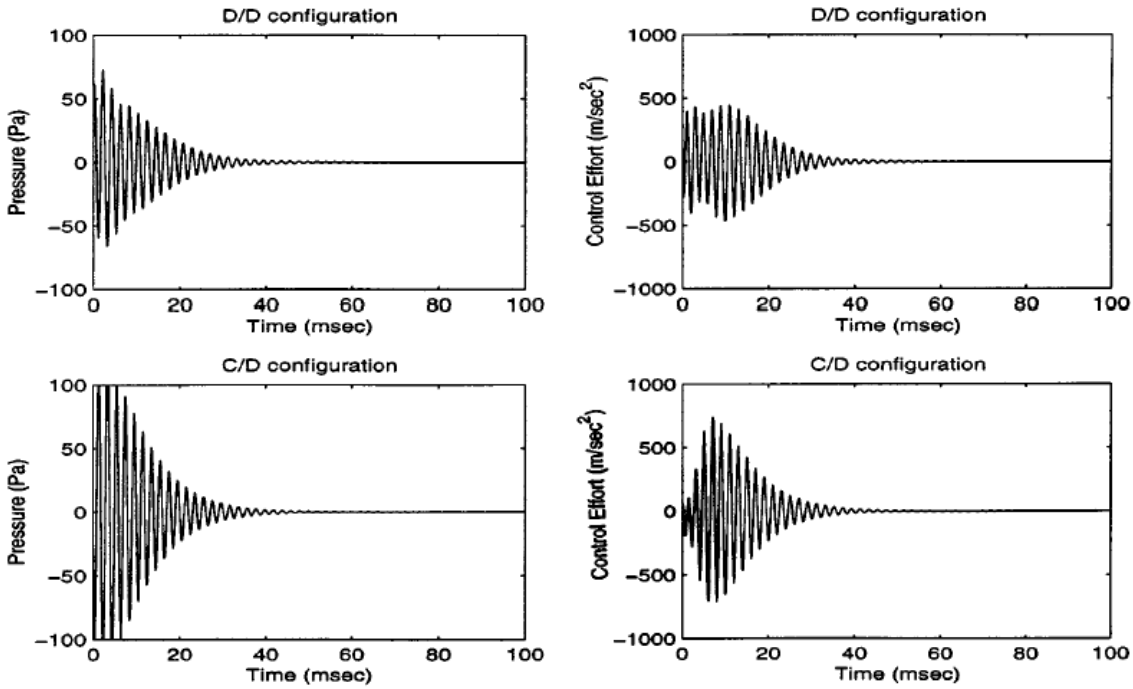


Figure 2-4: Theoretical results for LQG-LTR controller at two locations, ϕ=0.7

It can be seen that the experimental results match very closely with the expected theoretical results. One important advantage of this controller is that it does not induce any new instabilities. Since the controller compensates for the phase over the entire frequency range, no new instabilities are created by adding energy in phase with another frequency peak, since the phase does not crossover 0° and –360° as it does with a linear phase shifter. A disadvantage of the LQG-LTR controller is the inability to predict model robustness in advance, although some change in model robustness can be achieved by changing the gains in the cost function.

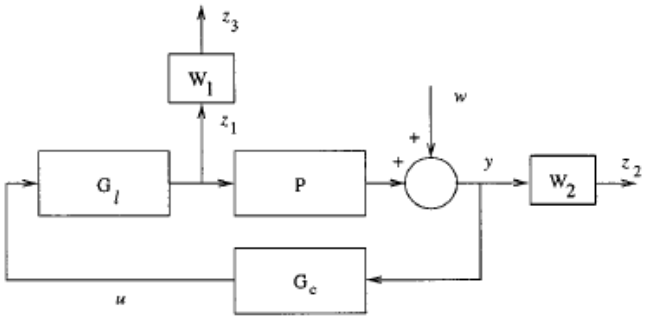


Figure 2-5: Block diagram of controlled combustor with H_∞ control

The next method of control is H_∞, which will ensure a chosen level of model robustness. A block diagram of the system is shown in Figure 2-5. A disadvantage of H_∞ control is the lack of a natural way to include time domain specifications. The weights W_1 and W_2 are chosen and iterated on to meet time domain requirements. To ensure model robustness with an allowable model error of ΔP, the weight W_1 is chosen such that

$$\|\Delta P(j\omega)\| < |W_1(j\omega)|$$

The controller designed in this way was found to be rather sluggish, as seen from the response in Figure 2-6. This controller was found to have a settling time of around 120 ms.

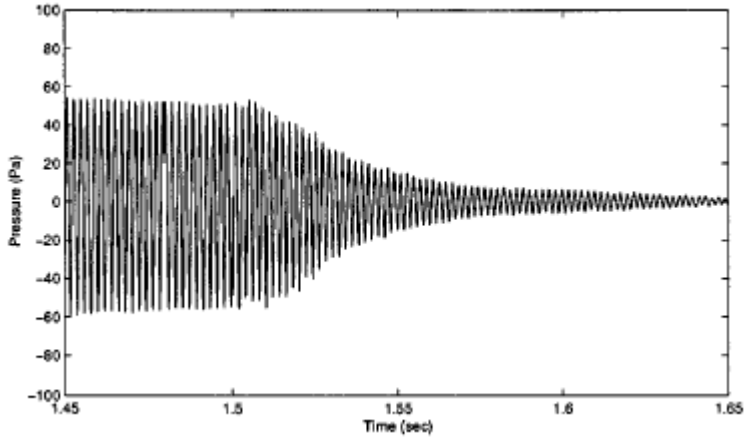


Figure 2-6: Controlled response of system with H_∞ control, φ=0.7

Model-based controller designs work very well, provided the system can be modeled. Errors in model design can cause further instabilities, and compensating with a controller exhibiting large model robustness can reduce the performance. It should be noted that this is still a linear method of control using a linear model of the system.

Mahmoud, et al. [8] did further model-based work was using Annaswamy's model. For this experiment, the same model as above was used to design a linear quadratic regulator (LQR) controller. First the actual uncontrolled response of the system was compared to the theoretical response of the system. The model predicts an unstable first mode at 300.1 Hz with a growth rate of 39 s^{-1}.

Using this model, an optimal controller was designed. This controller was designed to have approximately 180° of phase shift, and included a band pass filter to

eliminate all frequencies other than the instability. Sine sweeps were then performed on the actual system with compensator in place to provide data points to determine a transfer function for the combustor. This transfer function shows that the combustor contains an instability at 306 Hz with a growth rate of 44.5 s^{-1}, which is quite close to the predicted values shown above.

The experimentally determined model was then used to calculate an optimal LQR controller, which was able to eliminate the instability with a good settling time of approximately 60 milliseconds. Thus either a theoretically derived or experimentally determined linear transfer function is sufficient to design an LQR controller that will eliminate the instability.

2.2.5 Pulsed control

Although using a loudspeaker as an actuator to input pressure into a combustion system is a convenient way of conducting control experiments in the laboratory, it does not have much application in industrial gas turbines. Instability levels are in general too large to be controlled with a loudspeaker, and these actuators are not normally robust enough to withstand the heat and vibration inside an industrial combustor. Therefore, other methods of control have been proposed. One of the most promising is the use of secondary fuel injection to eliminate instabilities. However, proportional fuel injectors are not easy to implement and are not widely available. Therefore the use of standard, switched fuel injectors is desirable. This precludes the previous methods of control, however, which all rely on a variety of control input levels to achieve control. New methods of control relying on fuel pulses must be developed and the work done to this effect is outlined below. However, it may be seen that no investigation has been done on the frequency components of any of the pulsed control signals shown below.

Richards et al. [13] in 1995 examined the effects of secondary fuel injection on a premixed tube combustor with an instability at 300 Hz. Open loop tests were done by injecting fuel pulses from 50-80 Hz into the system with varying duty cycles. Initially control was fixed at 50 Hz and varying duty cycles were implemented. It was seen that duty cycles below ~35% would stabilize the flame (reduction of 10 dB) and higher duty

cycles would not. This can be explained somewhat by the control theory presented later in this thesis, in Section 3.1.

However, the experiment had a constant fuel flow rate to the secondary injectors of 0.067 g/s, or 14% of the total fuel flow, and a varying duty cycle. This indicates that the total amount of fuel entering the system is the same for all control signals, which consequently means that the height of the fuel pulse in addition to the pulse width must change for different control signals. (i.e. the integral under one period is constant) Mathematically this means that the gain is changed in two ways (see Section 3.3), and it is difficult to extract the method of control here. Lower duty cycle signals will thus have higher amplitude and may have a higher gain than higher duty cycle signals in this case. In addition, running in open loop ensures that there is not a constant phase shift, which likely explains the difference in control between one experiment at 40% and others.

The OH* spectrum was also examined for the varying fuel injection methods, and the fuel injection frequency could be seen in the spectrum. This indicates that secondary fuel injection is valid for modifying the heat release.

Tests were then done using an unspecified fixed pulse width and varying frequencies from 50-80 Hz. It was found that control was achieved regardless of frequency up to 80 Hz, where control was lost. This is possibly due to the control fuel interrupting the mechanism creating the instability and not due to out of phase addition of heat. This is not explained by linear control theory.

In general, this paper is valuable for providing a background for using fuel injection control. It also shows that a level of control (10 dB) can be achieved using only open loop control.

Heising et al. [4] in 2000 examined the effectiveness of modulating the primary fuel in the open loop to eliminate instabilities. Tests were done varying the frequency of fuel injection on a stable combustor rig from 170-600 Hz and the duty cycle from 20-80%. A main focus of this research was on the effects the various signals had on fuel droplet size and velocity. The fuel injector in this experiment was a pintle valve that pushed the fuel out into the combustion region.

The experiments concluded that the duty cycle of the control signal had a large impact on the size and velocity of fuel droplets. Smaller duty cycles produced smaller,

faster particles that burn farther from the injection point. It is interesting to note that at low duty cycles a larger pressure drop was seen across the fuel injection nozzle than at 75%. This could be equated to a higher amplitude pulse in a pulsed fuel injection signal, which would equate to more gain. In fact, the pulses in this system may appear more as triangular waves, with higher amplitude at the beginning of the pulse than at the end. It is important to note the changes resulting from using a fuel injection system compared to acoustic actuator. The CH* power spectrum reveals that the injection frequency can be seen in the heat release. As seen in Figure 2-7, it is also evident that harmonics of the system exist. It would be possible to build a time signal of the heat release pulse injected into the system from the fuel injector by adding the sine waves at these harmonic frequencies together. The direct effect on heat release (as measured by CH*) can be seen in Figure 2-8. There is some phase delay between the pintle opening and the heat release. It is also noted that the heat release shape is not exactly identical to the pulse shape.

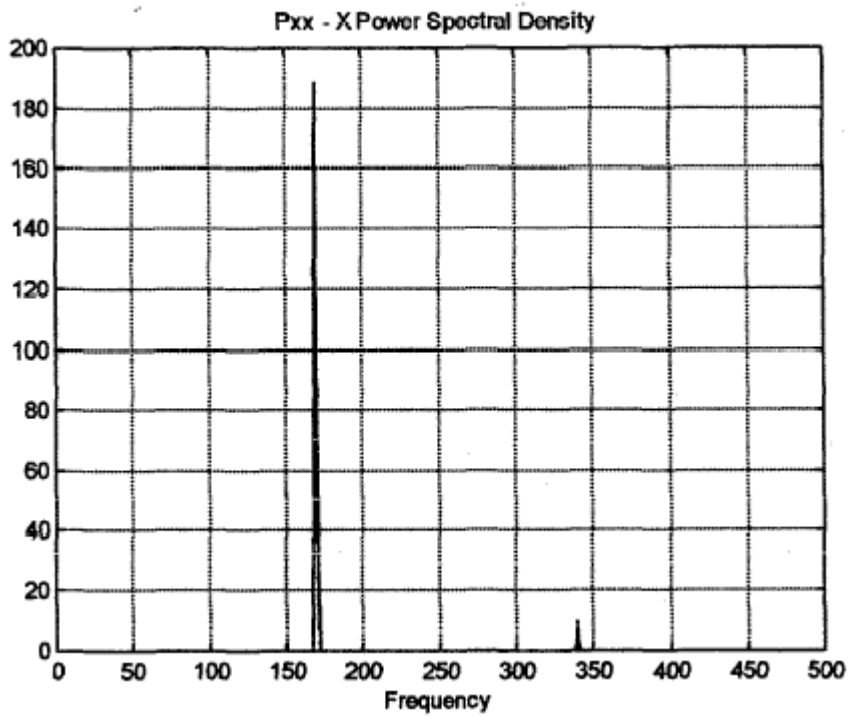


Figure 2-7: CH* spectrum from flame modulated at 170 Hz

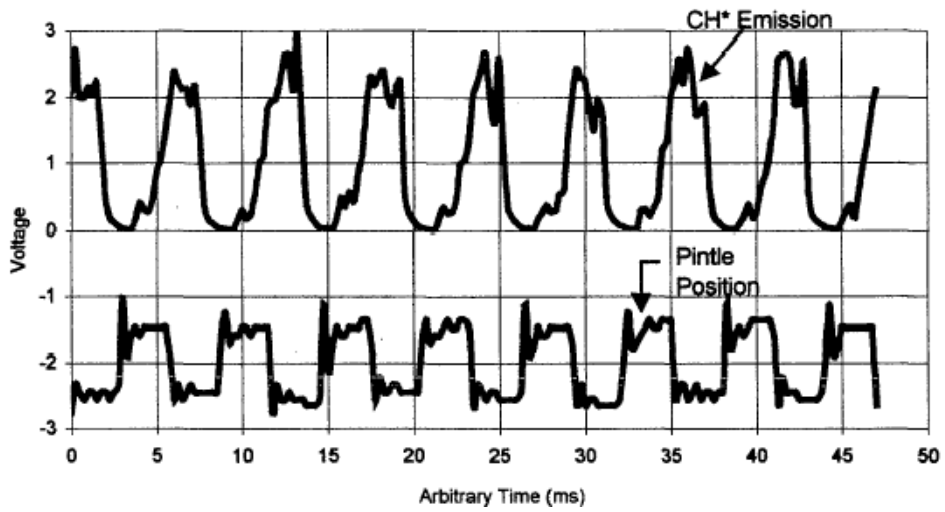


Figure 2-8: Time dependence of pintle displacement and CH* emission, 170 Hz modulated flame

It is stated that one other limiting factor is the time taken for the injector to open and close. As the duty cycle decreases and/or the frequency increases, this time is a greater percentage of the pulsed time and thus is thought to limit the effect of the fuel injection on the flame. Above 800 Hz the injector cannot be fully opened or closed in the desired pulse time. From a frequency perspective however, this is simply the bandwidth limitation of the fuel injector. The fuel injector is basically a low pass filter, in this particular instance at 800 Hz. Based on linear control theory, provided the instability is below the cut off frequency of the injector, control can be established. This is shown mathematically in Section 3.1 of this thesis. It will be shown that the lower control energy of a lower duty cycle control signal is not due to its shorter pulse width, but due to its lesser Fourier component at the instability frequency.

Finally, this paper examined the effectiveness of various control signals by finding the percentage of the injected fuel signal that was unsteady. It was found that lower duty cycle signals in general have a higher effectiveness, and it was stated that the effectiveness is affected by the duty cycle. Chapter 3 of this thesis will show that this is due to the fact that a more effective control signal will have a lower DC component. Only the non-DC components, in particular those at the instability frequency, act to damp out the instability. Fourier analysis can thus explain Heising's results mathematically as shown later in this thesis. Lower duty cycle signals are more effective because the mean of a lower duty cycle signal is closer to zero, and therefore the DC component of the signal is lower.

McManus et al. in 1997 [9] and further in 1998 [10] further examined pulsed control by utilizing a closed-loop secondary fuel injection control system. This system varied the width of the pulse proportionally to the height of the instability. As will be seen in Section 3.1, the gain of a pulsed signal is not linearly proportional to its pulse width. However, some interesting results of the McManus reference were found that can be further explained by examining the mathematical and control principles stated in this thesis.

An automotive (Bosch Jetronic) fuel injector was used for the control in this experiment. Initially open loop, non-reacting tests were conducted (using water for safety reasons) on the fuel injection system, and it was found that the fuel injector had a fairly flat frequency response over the range needed for the experiment, from 100-500 Hz.

An open loop frequency response of the reacting system using CH* measurements was then shown, and a peak at around 140 Hz indicates the instability. Other frequencies were input through the fuel injection system in this open loop configuration and produced peaks of approximately 20 dB at the forcing frequency, which indicates that the heat release can be controlled by this secondary fuel injector. A spectrum showing the unforced and forced cases is seen in Figure 2-9. Here the Fourier components of the pulsed signal at multiples of the 130 Hz forcing signal can clearly be seen.

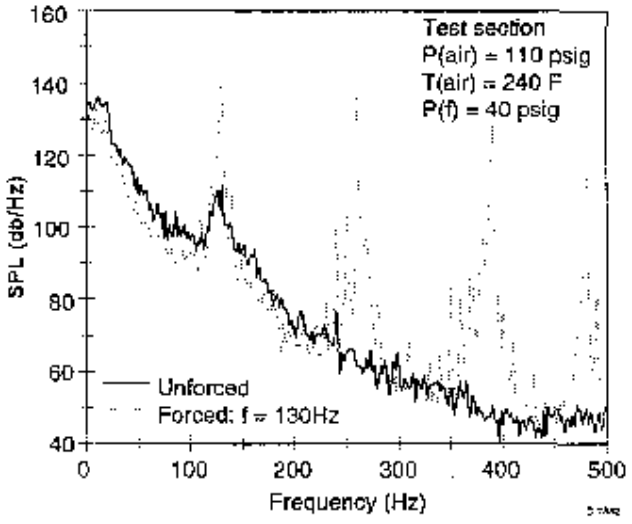


Figure 2-9: Open loop power spectrum with fuel modulation at 130 Hz

Closed loop experiments were then done with the control system seen in Figure 2-10. The phase delay is measured to the beginning of the fuel injection pulse, and it is noted that the best control is achieved by beginning the pulse just before the minimum of the pressure signal. As seen in Section 3.1 of this thesis, this is explained by the fact that the Fourier component's phase should be measured from the center of the pulse, and not the beginning. This is very significant because the control scheme implemented by McManus does not introduce a constant phase delay at the instability frequency. Since the pulse widths change proportionally to the instability level, it is seen that the phase shift of the controller also changes. Thus as the controller gains stability and changes pulse width with the new lower instability level, it may cause the control input to be at a non-ideal phase for control and thus lose stability instead.

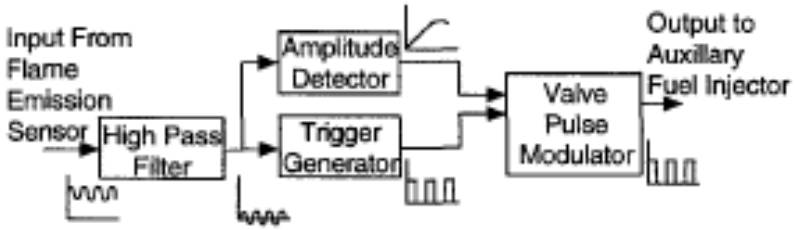


Figure 2-10: Control system for fuel injected pulse width modulation control

In addition, this controller, as seen in Figure 2-11, created secondary peaks. This is explained by the steep phase roll-off caused by the high pass filter poles, as shown above in 2.2.2, the analysis of a linear phase shifter.

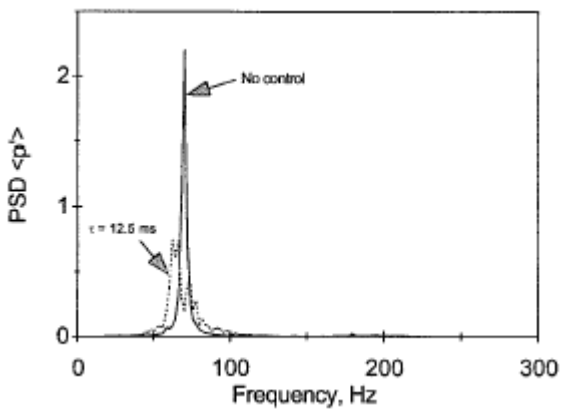


Figure 2-11: Secondary peaks generated by PWM controller

2.2.6 Subharmonic control

Pulsed control is normally achieved by sending pulses on zero crossings of the instability signal. Therefore, the next logical step is to leave out some of the pulses, and switch on every second or third zero crossing. This will result in a pulsed signal with a fundamental frequency at a subharmonic of the instability frequency. This method of control was examined by some researchers in a hope of gaining better fuel economy and reducing the bandwidth required of the fuel injector. Although the mathematics behind subharmonic control are identical to those for pulsed control, it has been presented as a separate topic in the literature and will thus be treated that way here.

Yu et al. in 2000 [16] examined subharmonic control of secondary fuel to reduce an instability of less than 100 Hz. Initially tests were conducted with a non-subharmonic pulse train to examine the effect of fuel droplet size on control. Two different sets of fuel injectors were used and the differences in control recorded. The first set of actuators (No. 1) were standard automotive fuel injectors and produced approximately 40 μm diameter fuel droplets, and was able to maintain control of the system using as little as 8% of the total average fuel flux. The second set of actuators (No. 2) were prototype pulsed fuel injectors with an air-assisted atomization mechanism. These injectors produced a finer mist of fuel and were able to maintain control using only 2% of the total fuel flux for control. This is shown in the paper to be predicted by the finer fuel mist of the No. 2 injectors. The tradeoff with using these injectors is a limited bandwidth of 150 Hz, which is still above the instability frequency in this rig.

The same combustion rig was then used to test subharmonic fuel injection. Two different methods of control were used. Initially tests were conducted using just closed-loop subharmonic control. It was found that, similar to a linear phase shifter, if control was added at the correct phase delay the system could be stabilized with a suppression of more than 10 dB of the instability.

According to the paper, it was found that by doing this the instability frequency would change enough that it would fall outside the range of the bandpass filter on the input to the controller and cause a loss of control. As seen in Saunders in 2.2.2, this is not the instability frequency shifting but rather secondary instabilities caused by a lack of proper phase control of frequencies other than the instability. This causes instabilities at

the phase crossover frequencies. To remedy this situation, the experimenters added a "pacemaker," or open loop pulsed signal, to the system. This resulted in an even greater level of control. In essence this is providing a new signal for the controller to phase lock to. Since the instability amplitude shrinks and a fixed height controller is used, the gain will increase as the system is controlled as seen in Section 3.3. However, using a pacemaker limits this increase of gain because it provides a lower limit for the instability amplitude. This lower gain prevents the instabilities from arising at the crossover frequencies of the filters and thus appears to give a greater level of control. It is also seen that subharmonic control with a pacemaker produces an even better level of control than a simple pulsed phase shifter. As will be seen in this thesis, this is due to the fact that the gain with a subharmonic signal is lower than a non-subharmonic signal. This reduction in gain further reduces the growth of secondary instabilities. It appears that the system with non-subharmonic fuel injection has too much gain, and is actually creating instabilities from the control signal.

Finally, the controller was modified further to vary the pulse width depending on the level of instability. It was configured in such a way that for instability levels below a certain level, no control fuel was injected at all. As will be seen later in this thesis, varying the pulse width does not linearly vary the gain of the controller. Thus this method is not certain to produce control. However, as is shown in the paper by Yu, control can be achieved with this method using a pacemaker. The gain of the controller can be properly controlled, and optimal instability suppression achieved, by turning off the secondary fuel control and using a pacemaker.

Both Jones et al. in 1999 [5] and Kim et al. in 2000 [6] from Penn State investigated using 4^{th} subharmonics to modulate the secondary fuel. Two different types of secondary fuel were investigated: natural gas and jet-A liquid fuel. The instability frequency was 336 Hz. As the fuel injector could only produce distinct pulses to 250 Hz, it was proposed to use subharmonic control. However, this did not necessarily mean the actuator did not have a response to 336 Hz and neither examined this effect experimentally.

There are several interesting phenomena from this experiment that can be explained by the mathematical principles in this thesis. For instance, in the case of

natural gas, a constant flow rate was used. Varying the duty cycle required a change in the time delay to maintain control. This is explained by the fact that the phase of a pulse needs to be calculated from the center of the pulse. Thus a fixed pulse height system with a fixed time delay to the start of a pulse will have a varying phase delay as the duty cycle changes. Conversely, the jet-A fuel was used with a fixed pulse width and a varying pressure, and no adjustments in phase were necessary. The fixed pulse width ensures a fixed phase delay, and the varying pressure is analogous to varying the pulse height, and thus the gain of the control signal.

It was also noted that as the pressure of the jet-A fuel is increased, control is increased. After reaching a pressure of 10 psig, however, an increase in pressure results in a decrease in control. This can be explained by the fact that the gain is most likely reaching its ultimate value, and thus the pulse train signal is replacing the instability as the limit cycle peak. This comment will be more fully appreciated after reading Sections 3.4 and 3.5

3 Theory

To fully understand why a pulse train controller works, the mathematics of the pulses must be understood. Since the instability is a resonance, which appears as a narrow band peak in the frequency domain, the time domain representation will be an almost pure sine wave. In practice the instability signal is filtered before entry into the controller to further enable the controller to lock onto the instability frequency by eliminating noise. A pulse train controller typically works by watching for zero crossings of this input signal. To achieve a phase delay between the input signal and the output pulse train, a time delay is added after the zero crossing before the controller outputs the pulse. This nonlinear controller thus does not have a fixed phase shift for all frequencies of the input signal. This will be important in further analysis. Variable pulse widths can be achieved by varying the on-time of the pulse. This means that the duty cycle of the output is also dependent on the input frequency, since a constant on-time and not a constant duty cycle is used.

This controller is nonlinear and does not have a linearization, since the output consists of frequencies not present in the input. However, since the acoustic plant significantly attenuates frequencies other than the resonance, only the output of the controller at the resonance frequency will be important. This is also a valid concept because as has been shown in the literature, linear control techniques are sufficient to eliminate the instability since control brings the system into the linear regime. Since the system can be treated as linear, superposition holds and thus only the output of the controller at the instability frequency is important.

To begin the analysis of the system, then, it is first important to find the component of the output signal at the instability frequency by performing Fourier analysis. It is then possible to examine the gain of the controller at the instability frequency using describing function analysis. This gain equation is then used to examine

both proportional height controllers, which will be useful in analysis, and fixed height controllers, which will be used in practice.

3.1 Fourier Series Expansion

The first step to examining the pulse train shown in Figure 3-1 is to perform a Fourier series expansion. It can be seen that the pulse train shown has a period MT_0, a pulse width w, and a pulse height X. If it is necessary to eliminate an instability with a period T_0 (and thus a frequency $f_0=1/T_0$), then a pulse train with a period of MT_0 will have a fundamental frequency of f_0/M, and will be called an M^{th} subharmonic pulse train. The positive and negative values of the pulse train reflect the fact that the controller designed for this application sends the pulse train to positive and negative values. This was done since a loudspeaker was used as an actuator. Fuel injectors will not need negative values.

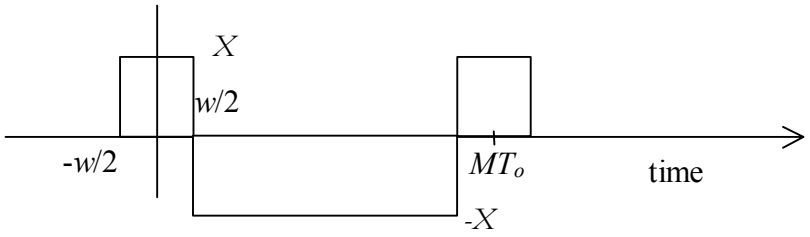


Figure 3-1: Typical pulse train

Since it is assumed in this case that the pulse train is symmetric about the y-axis, a cosine Fourier series will be used, which will be converted to sine waves later. Ignoring the DC term, we thus find that the Fourier series expansion in general is

$$f(t)=\sum_{n=1}^{\infty} c_n \cos\left(\frac{n\pi t}{MT_0}\right)$$

where the coefficients are given by

$$c_n=\frac{2}{MT_0}\int_0^{MT_0} f(t)\cos\left(\frac{n\pi t}{MT_0}\right)dt$$

It is first necessary to break the pulse train $f(t)$ into its three sections, and then evaluate the integrals.

$$c_n = \frac{2}{MT_0}\left[\int_0^{w/2} X\cos\left(\frac{n\pi t}{MT_0}\right)dt - \int_{w/2}^{MT_0-w/2} X\cos\left(\frac{n\pi t}{MT_0}\right)dt + \int_{MT_0-w/2}^{MT_0} X\cos\left(\frac{n\pi t}{MT_0}\right)dt\right]$$

$$= \frac{2X}{MT_0}\left[\frac{MT_0}{n\pi}\sin\left(\frac{n\pi t}{MT_0}\right)\Bigg|_0^{w/2} - \frac{MT_0}{n\pi}\sin\left(\frac{n\pi t}{MT_0}\right)\Bigg|_{w/2}^{MT_0-w/2} + \frac{MT_0}{n\pi}\sin\left(\frac{n\pi t}{MT_0}\right)\Bigg|_{MT_0-w/2}^{MT_0}\right]$$

$$= \left(\frac{2X}{MT_0}\right)\left(\frac{MT_0}{n\pi}\right)\left[\begin{array}{l}\sin\left(\frac{n\pi w}{2MT_0}\right) - \sin\left(\frac{n\pi\left(MT_0 - w/2\right)}{MT_0}\right) + \sin\left(\frac{n\pi w}{2MT_0}\right)\\ + \sin(n\pi) - \sin\left(\frac{n\pi\left(MT_0 - w/2\right)}{MT_0}\right)\end{array}\right]$$

By recognizing that for any integer n, $\sin(n\pi)$ will be zero, and combining terms, it is found that

$$c_n = \left(\frac{2X}{MT_0}\right)\left(\frac{MT_0}{n\pi}\right)\left[2\sin\left(\frac{n\pi w}{2MT_0}\right) - 2\sin\left(n\pi - \frac{n\pi w}{2MT_0}\right)\right]$$

Next it is necessary to evaluate the second term. For odd n, it is found that

$$\sin\left(n\pi - \frac{w}{2MT_0}\right) = \sin\frac{n\pi w}{2MT_0}$$

and thus c_n=0 for odd n. Conversely, for an even n, it is found that

$$\sin\left(n\pi - \frac{w}{2MT_0}\right) = -\sin\frac{n\pi w}{2MT_0}$$

Since the odd terms can be ignored, the substitution $n=2k$ can be made and the series evaluated for all integers k. Substituting, combining terms, and multiplying by w/w, it is thus found that

$$c_k = \left(\frac{MT_0}{k\pi w}\right)\left(\frac{2wX}{MT_0}\right)\left[2\sin\left(\frac{k\pi w}{MT_0}\right)\right]$$

Finally by remembering that $sinc(x)=sin(x)/x$, the coefficient can be found as

$$c_k = \left(\frac{4wX}{MT_0}\right)\text{sinc}\left(\frac{k\pi w}{MT_0}\right) \quad \textbf{(3)}$$

Thinking about things in the frequency domain, a pulse train signal will thus have components at multiples of the pulse train frequency determined by the coefficients of the Fourier series c_k. It should be noted that the coefficients are highly dependent on the duty cycle, which is given as $d = \frac{w}{MT_0}$. This relationship is nonlinear as well, and for certain duty cycle and subharmonic ratio combinations, will even result in a zero coefficient.

Finally, since the input will be thought of as a sine wave, it is desirable to rewrite the cosine expression as a sine wave, which will involve a phase shift of 90°. In addition, the controller will add a control time delay τ from the zero crossing to the beginning of the pulse. Since the Fourier analysis was performed to the center of the pulse, it is also necessary to shift the signal by the time $w/2$. Finally, the Fourier coefficient can be written as a gain and phase term, thus adding further phase shift to the sine wave signal. Thus the final representation of the output signal will be

$$\sum_{k=0}^{\infty}|C_k|\sin\left(\frac{k\omega}{M}(t-\tau-\frac{w}{2})+\angle C_k+\frac{\pi}{2}\right) \quad \textbf{(4)}$$

3.2 Describing Function Analysis

Now that an adequate description of the frequency components is in place, an expression for the gain of the system can be developed. Although the output cannot be linearized, describing functions may be used to calculate the gain of the controller. Describing functions are used to arrive at an expression for the gain and phase of a controller at a particular frequency. Since superposition holds, only the characteristics of the controller at the instability frequency are important. Thus describing function analysis is appropriate.

If the input to the controller is a thermoacoustic instability, it may be described as a pure sine wave of amplitude A at the frequency of instability ω, or $A\sin(\omega t)$. Depending on the particular implementation of the controller, the output (via the height of the pulse X) may be dependent on the input amplitude A. It will also be dependent on

the frequency of the input ω. Since it was seen that the phase delay will also depend on frequency, the output at the instability frequency can be written as the function $F(A, \omega)\sin(\omega t + \varphi(A, \omega))$. Thus the describing function is the gain and phase consisting of the output term over the input term,

$$DF = \left|\frac{F(A,\omega)}{A}\right| \angle\varphi(A,\omega)$$

To evaluate the describing function gain at the instability frequency, the Fourier component of the control signal at the instability frequency must be found. For an M^{th} subharmonic signal, the M^{th} component of the control signal will be at the instability frequency. Thus solving for C_k by setting $k=M$ will yield the equation of the sine wave at the instability frequency ω. Substituting in (1),

$$C_M = \frac{4wX}{MT_0}\operatorname{sinc}\left(\frac{wM\pi}{MT_0}\right) = \frac{4wX}{MT_0}\frac{\sin\left(\frac{wM\pi}{MT_0}\right)}{\frac{wM\pi}{MT_0}} = \frac{4X}{M\pi}\sin\left(\frac{w}{T_0}\pi\right) \quad \textbf{(5)}$$

Note from this equation that the component of the control signal that is seen at the instability frequency is proportional to the sine of the duty cycle (times π) and not the duty cycle itself. Therefore the signal will be at a maximum when the sine is 1, or when the pulse width is one half of the instability period. Also, the level of the control signal is inversely proportional to the subharmonic ratio M. Thus, as the subharmonic ratio is increased, the height of the pulse must be increased as well to inject the same amount of power into the system.

The above conclusions make sense from a Rayleigh criterion viewpoint as well. The Rayleigh criterion states that to eliminate the instability, the controller must extract sufficient energy from the system to prevent oscillations from growing and to instead damp them out. Therefore the controller must provide heat release in the negative half cycle of the pressure signal, which explains the ideal pulse width of $T/2$. In addition, the same amount of energy must be input to the system for all controllers. Thus if a pulse is only output every M^{th} cycle of the instability, the energy of each of these pulses must be increased by a factor of M.

The describing function, or the gain and phase of the system, can now be found as the ratio of the output signal over the input signal. It should be noted that this gain is valid at the instability frequency only and not at other frequencies. The phase of the input is taken to be the reference 0°, and thus the phase delay of the system is simply the phase of the output of the controller. The amplitude of the input is simply A at the instability frequency ω, and the amplitude of the output is given in equation (2). Thus the describing function is

$$DF = \frac{4X}{AM\pi}\left|\sin(\frac{w}{T}\pi)\right|\angle\left[\frac{\pi}{2} + \angle C_M - \omega(\tau + \frac{w}{2})\right]$$

It should be remembered that the delay τ is configurable in the controller. Thus any phase delay can be realized in the system. Recognizing this, the phase can now be ignored. For all following analysis, the delay time is set such that the phase delay at the instability frequency is identical for all cases. This allows analysis to be done by analyzing the gain of the controller only. Thus the gain of the controller is

$$DF = \frac{4X}{AM\pi}\left|\sin(\frac{w}{T}\pi)\right| \quad \textbf{(6)}$$

3.3 Fixed Height Pulses

To find practical ways to test the hypothesis that control is accomplished only through the component of the control signal at the instability frequency, it is necessary to compare the controller to a known condition. Thus, a linear phase shifter will be used as a control condition. The phase shifter is set to a phase determined to be optimal, and the linear gain is increased from zero until control is achieved. At each gain the level of the limit-cycle will be recorded. This curve will then be the baseline for examining pulsed control. This limit cycle amplitude curve can be thought of as a nonlinear limit, where the nonlinearities prevent the instability from growing further.

Initially fixed pulse height signals will be examined. For a particular pulse height X, subharmonic ratio M, and duty cycle d, the describing function can be plotted. To achieve this, values of A are substituted in the describing function and the gains recorded. These can then be plotted as amplitude A versus the linear gain X/A on the same axes as

the limit cycle amplitude, and a curve similar to the one shown in Figure 3-2 is created. The red dashed lines represent the theoretical describing function gains for two different pulse heights X_1 and X_2, and the blue solid line represents the experimentally determined limit cycle amplitude curve, with the stabilizing gain being the gain for the linear phase shifter where the instability is eliminated. In reality this gain will not be a constant but will have hysteresis characteristics.

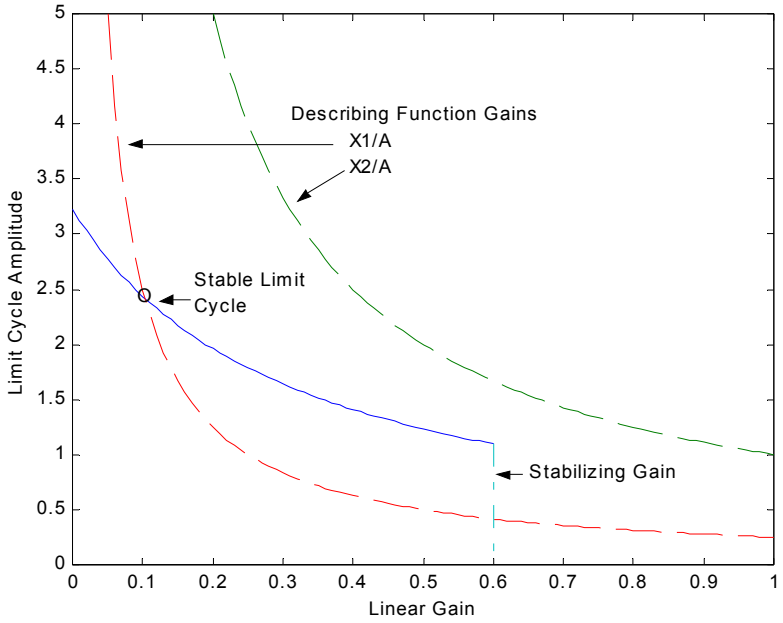


Figure 3-2: Fixed pulse height signal gain with linear phase shifter gain

3.3.1 Fixed pulse height controller with one intersection

The first part of the analysis will be for the lower of the two pulse heights, X_1. As the gain of the controller goes to zero, the amplitude of the instability tends toward infinity. In this case, seen on the figure for gains below the stable limit cycle with pulse height X_1, the nonlinear attributes of the system will dominate and the limit cycle amplitude will be the amplitude as shown by the linear phase shifter.

For gain regions above the stable limit cycle but below the stabilizing gain, there will not be enough gain to reduce the instability. The instability will grow towards the nonlinear limit, and the gain will consequently decrease until limited by the nonlinearities. Thus, for the region below the stabilizing gain, the nonlinearities will dominate and the system will stabilize at the intersection of the describing function gain

and the limit cycle amplitude curve. The name "Stable Limit Cycle" is given to this intersection point due to the fact that the system will remain at this point if it is there, and will be drawn to it from either side if the system is not.

Finally, it can be seen that as the limit cycle amplitude approaches zero and the gain of the controller passes the stabilizing gain, the gain of the controller will continue to increase and eventually approach infinity. The instability will thus be eliminated. However, practical systems will not allow a gain of infinity, and will be limited by other aspects. These will be seen in Section 3.5 where ultimate gain is discussed.

3.3.2 Fixed pulse height controller with no intersections

For the second pulse height X_2 in Figure 3-2, there is no intersection between the describing function gain and the limit cycle amplitude curve, and the describing function gain is always greater than the linear phase shifter gain for a particular limit cycle amplitude. Therefore adequate gain is available at all times, and the system is always stable. However, as the system is stabilized and the limit cycle amplitude decreases, the gain will once again tend toward infinity and will be limited by the ultimate gain as in the first case.

It can be seen in this instance that after the gain passes the stabilizing gain and continues to increase, there is an excess of gain in the system, which can be seen as wasted energy. With a known limit cycle amplitude vs. gain curve, it should be straightforward to compute values for the pulse height X, the pulse width w, and the subharmonic ratio M to ensure adequate gain for stabilization. However, after stabilization has been achieved, these parameters may be modified to shift the describing function curve down to limit the energy wasted. For instance, after stabilization has been achieved, the subharmonic ratio M could be increased to limit the gain and also the number of actuator cycles. Another possibility is to reduce the pulse width, and thus the duty cycle, to use less fuel in a fuel injected controller. However, care must be exercised in this case as this is not a linear relationship.

3.3.3 *Fixed pulse height controller with two intersections*

There is a third possibility for the fixed pulse height signal, as is seen in Figure 3-3. In this case there are two intersections of the limit cycle amplitude curve with the describing function gain. For gains below the unstable limit cycle gain, the system will behave identically to the two cases above. However, for gains above the unstable limit cycle, there will be excess gain in the system. This will cause the amplitude of the instability to decrease, and thus the gain to increase. When the gain increases past the stabilizing gain, it will continue to increase as before. In essence this system can achieve control with a slightly lower gain due to the accelerating effect of the unstable limit cycle. This intersection is given the name "Unstable Limit Cycle" due to the fact that the gain can not be fixed at this point. If the gain is lower than the unstable limit cycle gain, the system will accelerate towards the stable limit cycle. If the gain is higher than the unstable limit cycle, the gain will accelerate towards infinity. Thus the limit cycle repels the controller, and is termed "unstable."

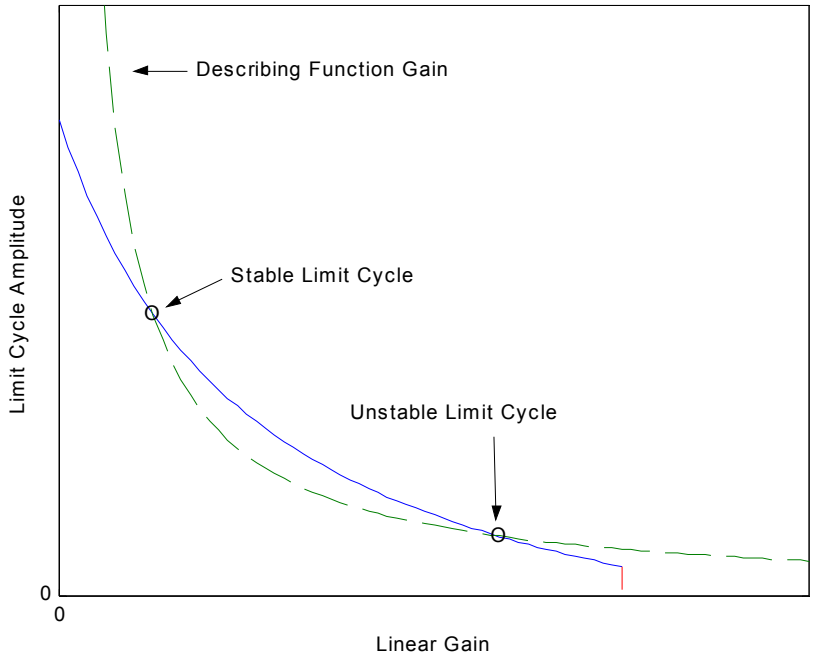


Figure 3-3: Describing function gain with stable and unstable limit cycles

3.4 Proportional Height Pulses

The second type of signal, which will be very important experimentally for proving the concepts shown here, is the proportional height pulse. For this signal, the amplitude of the instability A is found and the height of the pulse set as a multiple k of this amplitude. Therefore

$$DF = \frac{4k}{M\pi}\left|\sin(\frac{w}{T}\pi)\right|, \qquad k = \frac{X}{A}$$

It can be seen that in this case, for a constant subharmonic ratio and duty cycle, that the describing function gain will be a constant. This will be represented by a vertical line on the chart, as seen at the upper end of the describing function gains in Figure 3-4. In practice, the height of the pulse that can be generated will be limited by the control system. When the pulse height reaches its maximum, the controller will behave like a fixed pulse height controller. This is represented by the lower part of the describing function gains in Figure 3-4. A proportional height signal can thus be seen as limiting the maximum gain of the controller.

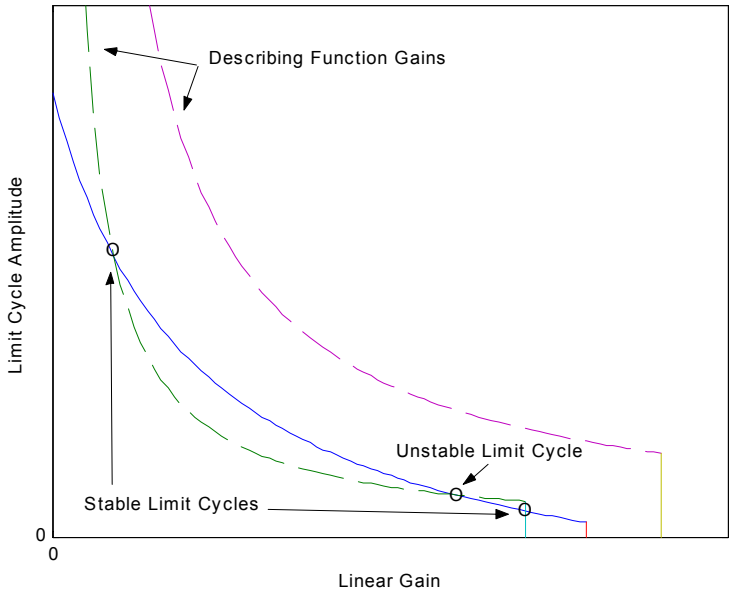


Figure 3-4: Proportional height pulse describing function intersections

3.4.1 Proportional height controller with three intersections

In the case seen above, it will be noted that the constant gain section of the lower describing function crosses the limit cycle amplitude curve. A stable limit cycle will be generated at this point. Since the gain cannot increase, the limit cycle amplitude will be fixed at the level shown for that gain *k*. The instability cannot be reduced any further for that gain with this controller.

It should be noted that there are two stable limit cycles separated by an unstable limit cycle. In this case, the state of the system when the controller is turned on is important. If the combustor and the controller are turned on simultaneously, the limit cycle amplitude will initially be below the unstable limit cycle amplitude. The instability will grow and stabilize at the stable limit cycle on the right. However, if the combustor is already running and the controller is turned on, it will not have enough gain in the system to eliminate the instability. In this case the system will stabilize at the stable limit cycle to the right. Once again, the unstable limit cycle can be seen as being repulsive. For lower gains than the unstable limit cycle, the system will be driven to the lower stable limit cycle. For higher gains, it will be driving to the upper stable limit cycle.

3.4.2 Proportional height controller with no intersections

The upper describing function of Figure 3-4 can now be examined. In this case, the constant gain section of the describing function is greater than the stabilizing gain, and the rest of the describing function contains enough gain at each point to drive the instability lower. In this case, the system will always be stabilized, and the gain will be capped at the constant gain. This situation is ideal as it will prevent the instabilities that arise from an infinite gain controller as seen below in Section 3.5, and it will always eliminate the instability.

3.5 Ultimate Amplitude

Although it could be assumed that once the original instability was eliminated the system would be stabilized, this is not strictly true. As the instability is reduced, the gain of the controller increases. For a fixed amplitude pulse and fixed pulse width, the

controller is inputting a fixed amount of energy to the system. This energy will in general be more than is required to maintain stability. Since the controller is producing energy at the original instability frequency, a new frequency peak in the spectrum will grow, induced by the controller. In a sense, the extra energy emerges as a new limit cycle and it is not possible to achieve a stabilized state having zero pressure oscillation.

From a controls standpoint, the instability can be seen as a pole crossing from the stable left half plane into the unstable right half plane. As the gain is increased, the original unstable poles are driven back into the stable left half plane. However, as the gain continues to increase, other system poles such as those from filters will be driven to the unstable regime, as seen below in Figure 3-5. The acoustic pole is a characteristic of the physical parameters of the tube. With the addition of heat release, the pole becomes an unstable thermoacoustic pole. The controller is able to move this pole towards the stable regime, achieving stability at k_{stab}. With further increases in the gain, other system poles will move to the right, and at the ultimate gain, k_{ult}, will become unstable. The frequency of this instability may not be equal to the original instability frequency and will be dependent on the frequency of the poles that become unstable.

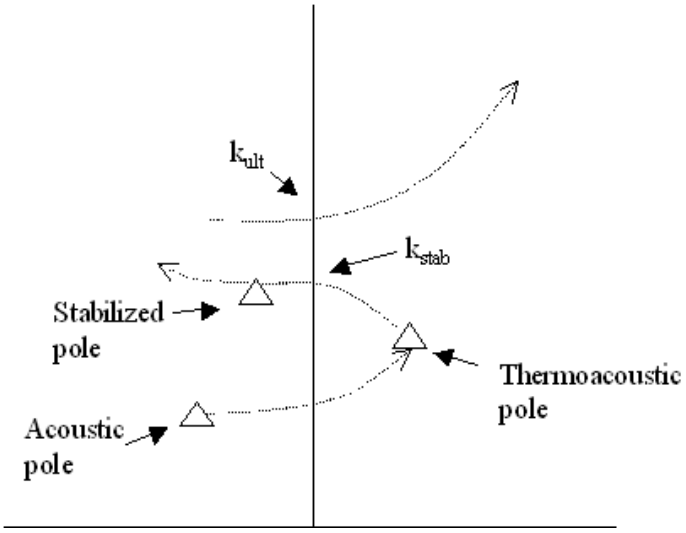


Figure 3-5: Pole movement in the complex plane with increasing gain

The ultimate gain can be found by using a linear phase shifter and increasing the gain until the system becomes unstable. For a fixed pulse height controller, as the gain increases past k_{ult}, the instability will again grow. This will cause a decrease in gain, and bring the pole back to the imaginary axis. As a result, a stable limit cycle will be

produced at k_{ult}. The level of this limit cycle can be calculated from the equation for the gain shown in (4). A 50% duty cycle will be assumed for this case, causing the sine term to equal one.

$$k_{ult} = \frac{4X}{AM\pi} \quad \textbf{or} \quad A = \frac{4X}{k_{ult}M\pi}$$

Therefore the amplitude of the ultimate limit cycle will be proportional to the pulse height X and inversely proportional to the subharmonic ratio M. Thus, although the system can never be stabilized to zero pressure due to the pulsed nature of the controller, the amplitude of the limit cycle can be controlled by the parameters of the controller.

4 Experimental Results

This section will show that the results for a real combustor rig match with what has been theoretically predicted. First, a linear phase shifter is examined to provide a baseline. Proportional subharmonic signals are then used to prove the hypotheses in Chapter 3, and then these results are used with fixed pulse height signals to show that the limit cycle height can be predicted.

4.1 Terminology

In this paper a subharmonic signal will be referred to as a *1/M x%* signal, where *M* is the order of the subharmonic and *x* is the duty cycle of the control signal. Therefore a 1/3 signal will be a subharmonic signal with a pulse every third zero crossing of the pressure signal. Note that the duty cycle is the percentage of time that the *control* signal is high. Therefore a 1/2 25% signal will have the same pulse width as a 1/1 50% signal, but will pulse only every other period of the limit cycling pressure signal. The control signal is filtered just above the instability frequency so that only subharmonic components and components at the limit cycle frequency are considered. The magnitude of the resulting filtered 1/1 signal will be determined by the first Fourier component, which will depend on the duty cycle. Examples of 1/1, 1/2, and 1/3 signals with equal pulse widths are shown in Figure 4-1, along with the limit cycling pressure signal. It is clear that the Fourier components will be different in each of the three cases.

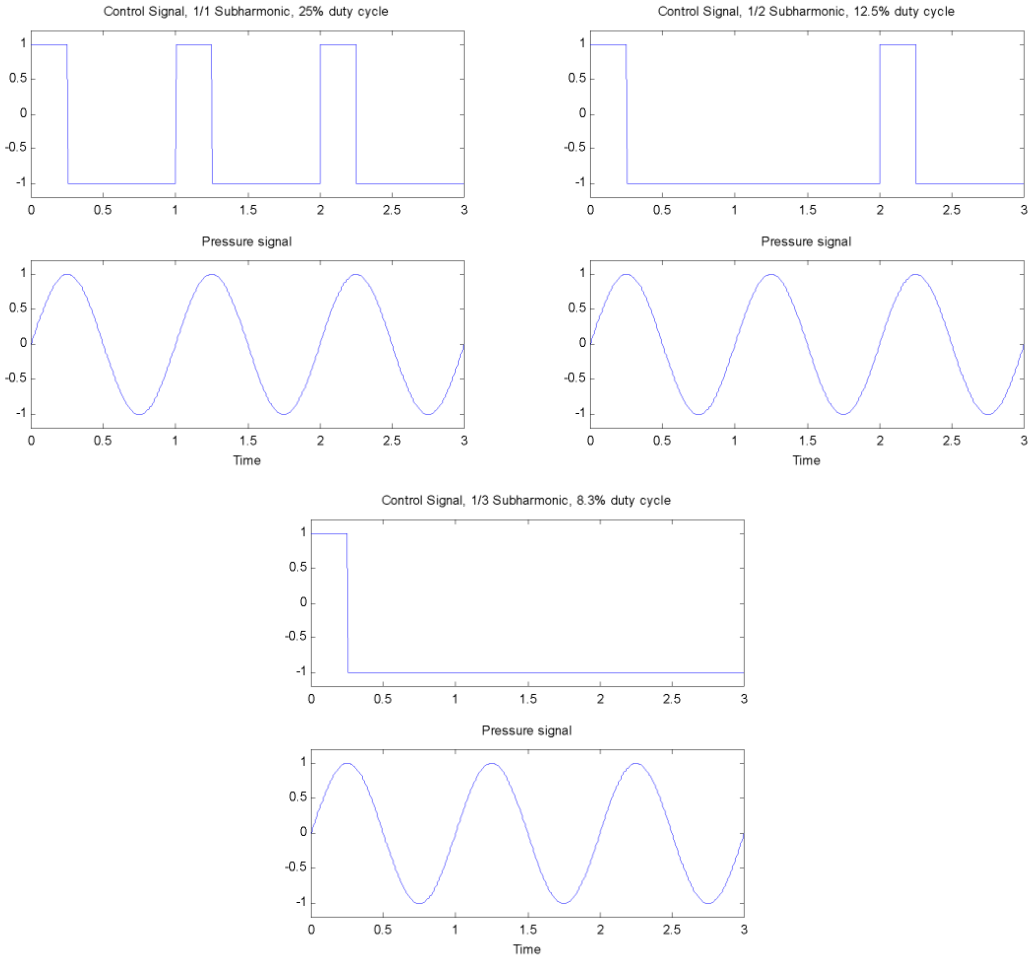


Figure 4-1: Typical subharmonic control signals

4.2 Simulation

It is now possible to test the theory using simulation via Matlab's Simulink. The simple model shown in Figure 4-3 was used for this purpose. The C code written for the DSpace board (see Appendix A) was converted to Matlab code and used to generate the pulsed control signal. A graphical user interface was also built to control the system and is shown in Figure 4-2. This GUI allows the user to set the number of samples to delay after a zero crossing is detected before triggering the pulse, the number of samples to hold the signal high (pulse width), the gain of the controller k, and the subharmonic ratio M. From these the phase delay in degrees and the duty cycle are calculated, as well as the frequency of the instability. In addition, the amplitude of the control signal and the amplitude of the instability are shown. It can be seen that for this particular example the

controller has a 42 sample delay, which equates to 45° of phase delay, and the signal is held high for 339 samples, or an equivalent duty cycle of 48%. The subharmonic ratio is 2.

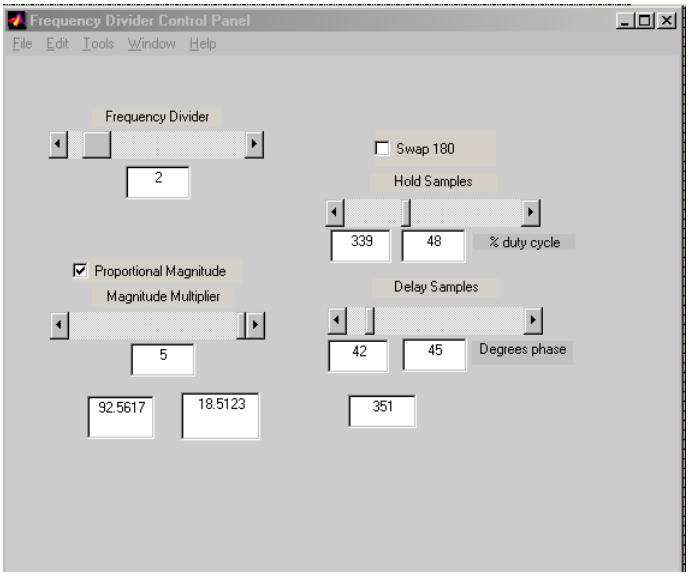


Figure 4-2: Matlab GUI for control of system

The acoustic resonance was modeled as a lightly damped resonance at 178 Hz, which is equivalent to the resonance in the experimental tube, and the heat release is modeled as a 2nd order low pass filter with a break frequency of 178 Hz. The gain of the acoustic resonance was then set such as to cause an instability. The nonlinearity in the system was modeled as a simple saturation nonlinearity clipping the signal through the heat release at ±5. The controller was added as a feed-forward controller acting through the acoustic plant, as will be the case with an acoustically controlled system. The pressure of the system, the heat release, and the control signal can all be monitored.

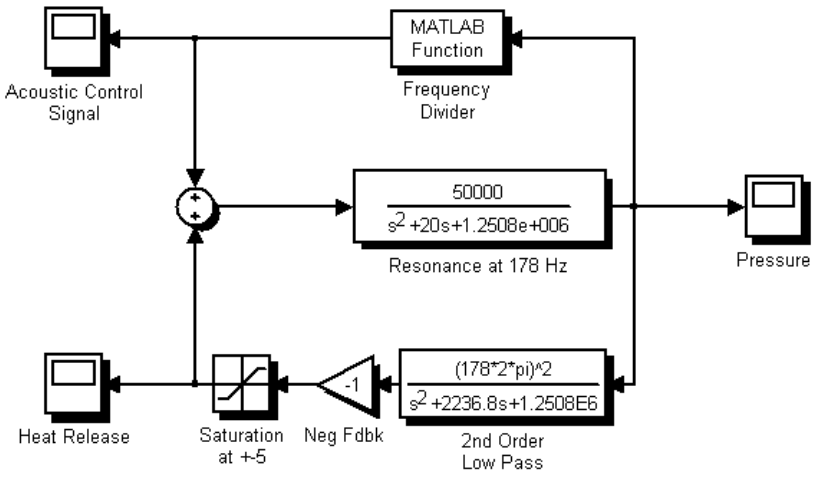


Figure 4-3: Simulink block diagram for modeling

The control signal, heat release, and pressure for a 1/2 50% duty cycle control signal (45° phase delay) is shown in Figure 4-4. It should be noted that the second Fourier component of a 50% duty cycle signal is 0. Thus, this control signal has no component at the instability frequency (178 Hz) and cannot reduce the heat release below the saturation limits or eliminate the instability, even with a gain of 5. Other values of phase delay were tried as well and did not change the simulation results. This agrees very well with the expected results from the theory.

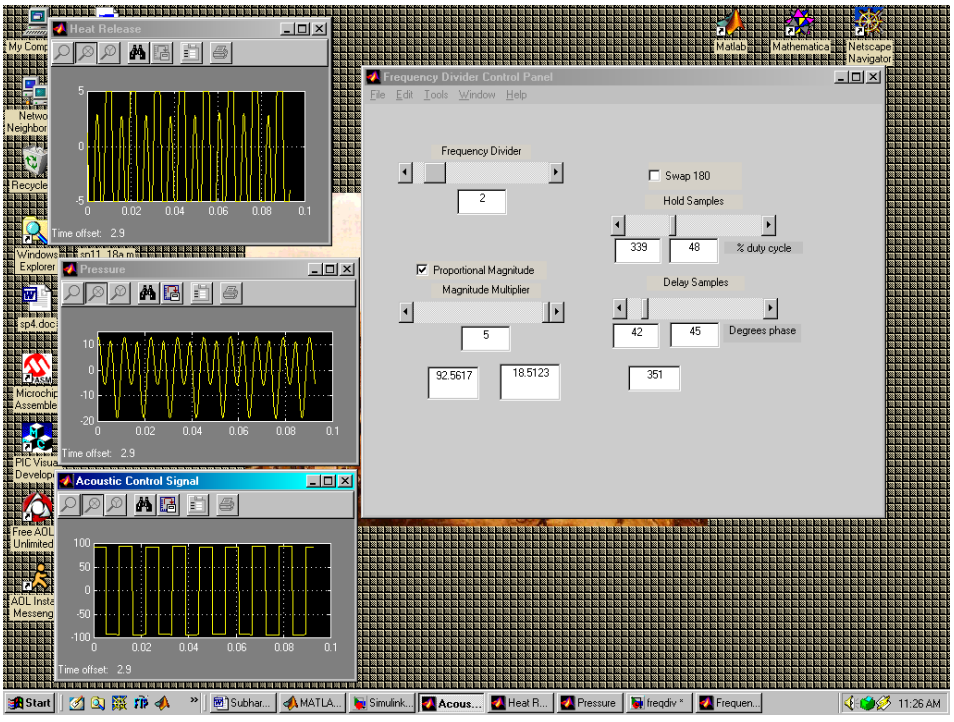


Figure 4-4: Simulink model unable to gain control with 50% half-harmonic

Control can be established by changing the signal to a 1/2 33% signal as seen in Figure 4-5. The control signal will then have a Fourier component at the fundamental frequency. It can be seen that the heat release drops below saturation and the pressure reduces to zero. Note that no changes in the gain or the phase were made. This provides further evidence that the duty cycle of the control signal is a crucial aspect of the controlled system.

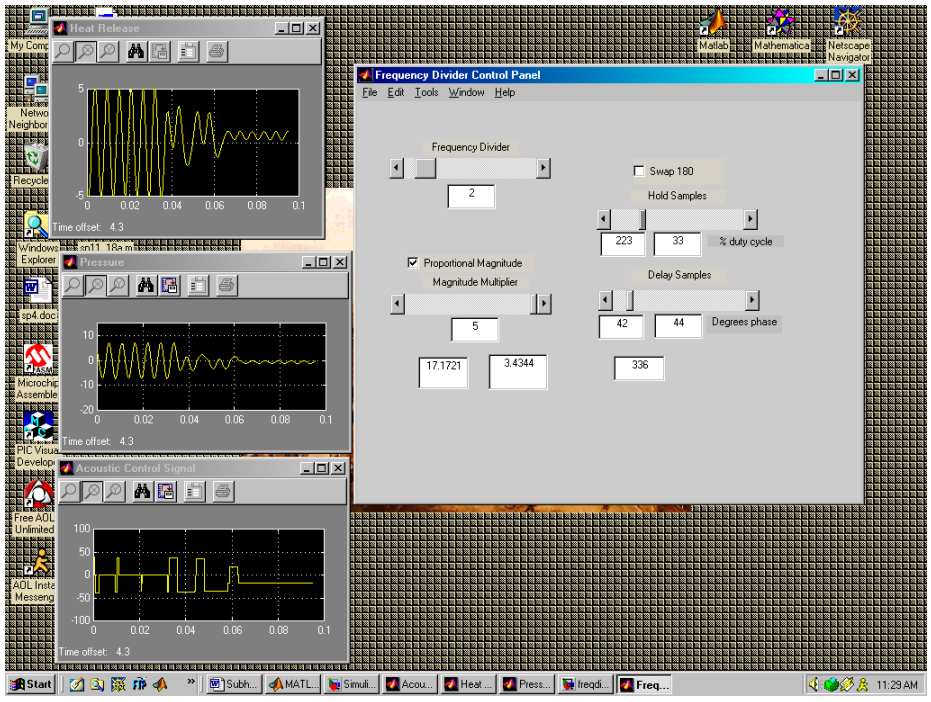


Figure 4-5: Simulink model able to gain control with 33% half-harmonic

To find the required gain to stabilize the thermoacoustic model of the system, a root locus of the linear system was generated, ignoring the nonlinearity. The controller was modeled as a simple phase delay, since describing function analysis as in Section 3.2 indicates that the gain at the instability frequency is only dependent on the control component at that frequency. A wide range of phases will cause the unstable poles to cross the imaginary axis into the left half plane. For the example shown in Figure 4-6, a phase delay for the control signal of 78° was chosen. This example uses a 1/2 30% control signal, and a 2nd order Pade approximation for the phase delay was used in place of the controller for computing the root locus. Note that the root locus does not depend

explicitly on the subharmonic M or the duty cycle, only the effective phase. It should be noted that to achieve the phase delay of 78°, an indicated phase delay as seen on the GUI of 60° must be used. This is due to the fact that the control algorithm calculates phase delay to the beginning of the pulse, whereas the delay of interest is to the center of the pulse, or the maximum of the sine wave component of the pulse.

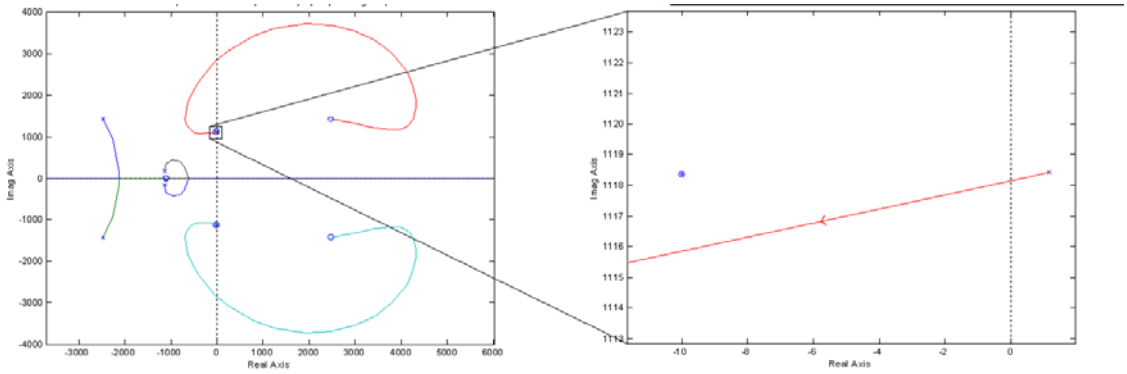


Figure 4-6: Root locus for thermoacoustic model showing poles becoming stable

The gain required to stabilize this system is the gain where the locus crosses the real axis, and can be found to be 0.054.

From the simulation it was found that for this case the minimum gain of the subharmonic controller that would steadily reduce the instability was 0.089. This was determined by watching the amplitude of the instability in the GUI and recording the gain of the controller that caused the amplitude to steadily decrease. The second Fourier component for a half harmonic 30% can be calculated as

$$c_2 = \left(\frac{4wX}{MT_0}\right)\text{sinc}\left(\frac{k\pi w}{MT_0}\right) = 4(0.3)X\frac{\sin[2\pi(0.3)]}{2\pi(0.3)} = 0.6055X$$

The subharmonic gain to stabilize the system, 0.089, is the ratio of the height of the pulse to the height of the instability (*X/A*). Therefore the overall linear gain at the instability frequency can be calculated by setting *X* to 0.089. Thus

$$k_{linear} = c_M k_{subharmonic} = (0.089)(0.6055) = 0.054$$

This gain can also be found directly from the describing function as

$$DF = \frac{4X}{AM\pi}\left|\sin(\frac{w}{T_0}\pi)\right| = \frac{4X}{AM\pi}\left|\sin(\frac{Mw}{MT_0}\pi)\right| = \frac{4(0.089)}{2\pi}\left|\sin\left(2\pi(0.30)\right)\right| = 0.054$$

Since this is the gain expected from the root locus using a simple phase shifter at the frequency of instability, the simulation provides evidence that the mechanism of control for a subharmonic pulse is identical to that of a linear phase shifter. Thus the component of the subharmonic signal at the limit-cycle frequency is indeed the controlling signal.

4.3 Experimental Setup

4.3.1 Hardware

The combustor used for the experiments is shown in Figure 4-7. The combustor is a steel tube that is acoustically closed at the bottom and open at the top. Premixed methane and air are injected at the bottom and a ceramic honeycomb flame holder is located at the tube midpoint (where the window can be seen in the picture).

Figure 4-7: Rijke tube combustor

The second acoustic mode of the tube, at a frequency near 180Hz, goes unstable for a wide range of equivalence ratios and flow rates [14]. For the control experiments a total flow rate of fuel and air of approximately 130 cc/s was used, with an equivalence ratio φ=0.55.

A block diagram of the experimental system is shown in Figure 4-8. The pressure transducer signal in the combustor is band-pass filtered before the A/D conversion to eliminate spurious zero crossings. The D/A output is filtered to protect the speaker, as a speaker is not mechanically designed for pulse inputs. However, all subharmonics are allowed to pass into the combustion system. The signal is sampled at 10 kHz to allow for as great a resolution as possible in setting phase and duty cycle. This gives approximately 55 samples per period for a 180 Hz signal. Phase and duty cycle are set as a number of samples, so smaller sampling periods are desired for greater resolution.

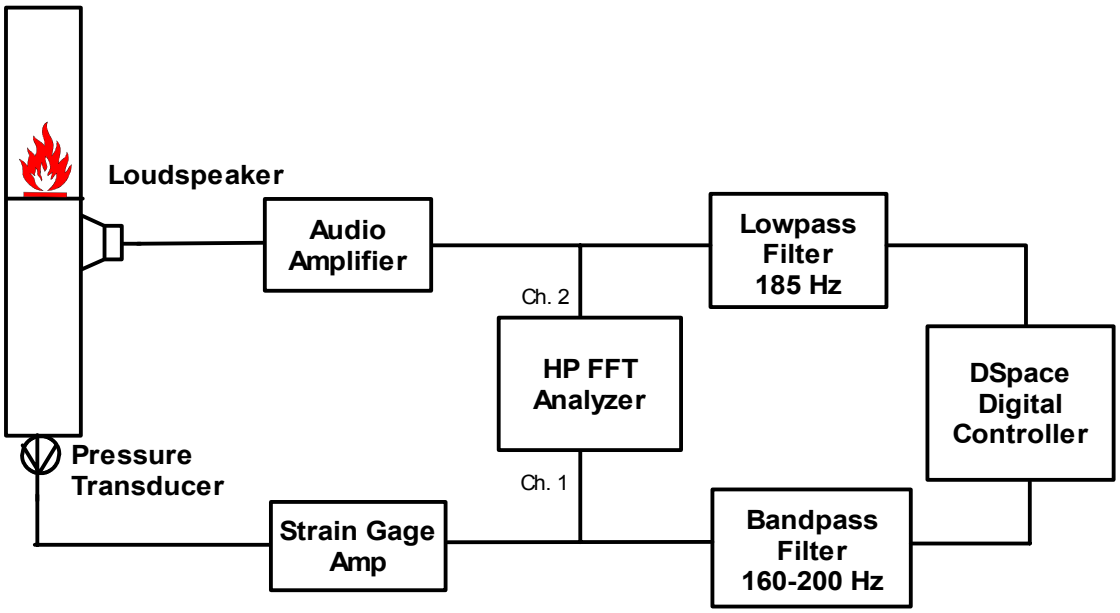


Figure 4-8: Test system layout

The DSpace DS1103 DSP board contains 20 analog inputs and 8 analog outputs. The first 16 inputs are multiplexed across four 16-bit A/D converters, with each converter containing four inputs. Converter one contained channels 1-4, converter 2 channels 5-8, and so on. The final 4 inputs each have their own 12-bit A/D converters. Since at most two inputs were needed for the controller, it was possible to use the 16-bit converters without encountering any timing issues due to multiplexing. Channel 1 was used for the

primary input channel, and channel 5 was used as the secondary input channel to add pacemaker signals and other desired signals. Initially the 12-bit channels were used. However, these limited the amount of control that could be achieved, as when the amplitude of the instability went below the threshold of 1 bit, the controller could no longer detect any zero crossings and therefore no control signal was output. Approximately 10 dB more reduction in the instability was achieved using the 16-bit converters over the 12-bit converters. The eight outputs are all 14-bit D/A converters, and output channel 1 was used as the primary output of the controller.

The configuration of all other hardware was kept as constant as possible throughout all experiments to prevent skewing of any data due to differences in the gains or phases of the individual components.

The pressure transducer was first run through a Measurements Group 2120A strain gage amplifier with a gain of 10 and a multiplier of 20, for a total gain of 200. This input to the system was then band pass filtered by sending it through a DL Instruments 4302 analog low pass filter set at 200 Hz and an identical filter configured as a high pass filter at 160 Hz. The output of the controller was low pass filtered by an Analog Devices digital filter set at 185 Hz. The audio amplifier used was a Peavey CS 800X. Identical gain settings of +6 detents on the volume knob were used for all experiments. The DSpace DSP board was installed in a Micron 650 MHz Pentium III PC with 128 MB of RAM running the Windows 2000 operating system.

All data was recorded with an HP 35665A Dynamic Signal Analyzer. In most cases the frequency range of interest was set from 0-800 Hz. All power and linear spectrum measurements, including limit cycle levels, were recorded after a 30 average sample was taken. This sample took approximately 30 seconds. Thus the data can be thought of as an average of 30 seconds.

4.3.2 Software

The experimental algorithm was written in C and implemented on a DSpace DS1103 controller. The source code is shown in Appendix A, and a detailed explanation of the operation of the system is given in Appendix B. A brief overview will be shown here. The algorithm generates a subharmonic pulse train in the time domain, which is

based on detecting zero crossings of the input signal. In the case of the tube combustor, the combustor pressure fluctuation is used as the input to the controller.

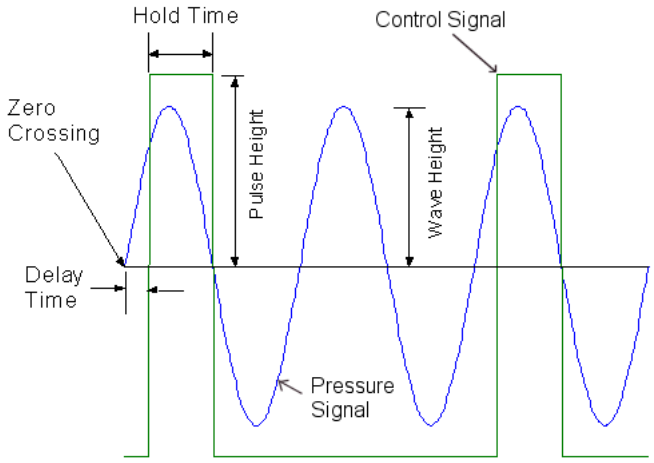


Figure 4-9: Control algorithm illustration

As shown in Figure 4-9, the algorithm starts a timer when it sees the previous sample of the input less than zero and the current sample greater than zero. When this timer reaches the value "Delay Time," a user selectable number of samples, it will set the control signal high. The height of the pulse can be fixed or made proportional to the wave height with a user-selectable gain. When the pulse is initiated, the hold timer is started. When this timer gets to the "Hold Time" number of samples, the pulse is ended and the signal goes to the negative value of the height calculated previously. The algorithm then returns to detecting zero crossings. When the set number of zero crossings M is reached (M=2 in Figure 4-9) the delay timer is started again and the next pulse is output. The algorithm, in addition to generating the pulse train, calculates the approximate frequency of the input signal, the approximate phase delay of the output, and the approximate duty cycle of the output. This allows different signals to be easily compared. By counting the number of samples between zero crossings and multiplying by the sample period, the period of the input can be calculated. The hold time can then be calculated as the number of hold samples multiplied by the sample period, and thus the duty cycle (referenced to the subharmonic signal) is

$$\text{duty cycle} = \left(\frac{\text{hold time}}{M\,(\text{input period})} \right) 100\%$$

In addition to being able to output subharmonic signals, the control code also contains a standard linear phase shifter. This is simply a time delay and gain applied to the input of the controller, as explained in Section 2.2.2.

The controller is configured via a graphical user interface designed in DSpace's ControlDesk 2.0 software. The design and operation of this interface is explained in detail in Appendix B. This interface allows the user to configure the number of samples to delay and the gain for the linear phase shifter. For the subharmonic controller, as is seen in Figure 4-10, the number of delay samples as well as number of hold samples can be configured. In addition, the subharmonic ratio (frequency divider) and gain or fixed pulse height can be configured. The controller calculates the frequency of the input, the phase delay in degrees to the beginning of the pulse, and the approximate duty cycle of the output. In addition, the GUI displays the input and output signals for reference. To the right of the screen are other switches and displays used for experimental variations. For instance, a second input may be read and added directly to the output. This input is shown in the lower right of the screen.

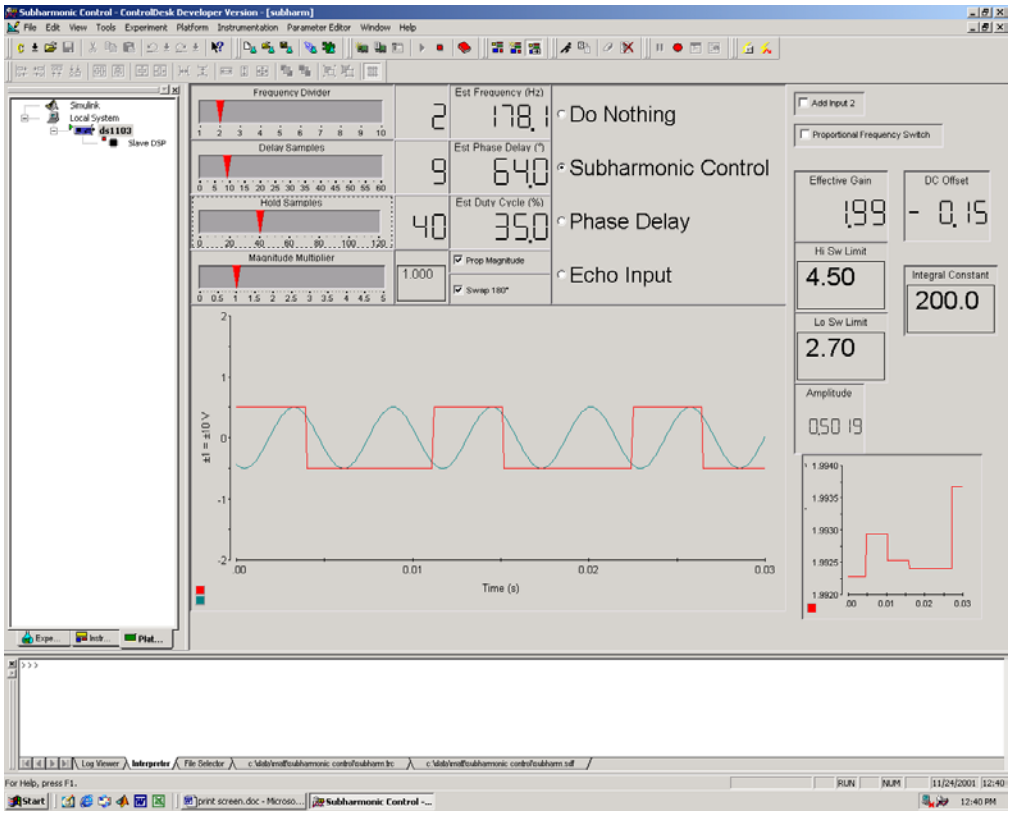


Figure 4-10: ControlDesk GUI showing subharmonic controller operating

4.4 Linear Phase Shifter Results

4.4.1 Hysteresis curve

The linear phase shifter simply outputs the input signal after a delay, thus passing all frequency components of the input (within the sampling rate and filtering constraints) to the output. Experiments on the combustor were first done using a fixed gain linear phase shifter to determine the optimal phase shift to be used. This phase shift was then used for all subsequent phase shift experiments. It was found that a delay of ~288° (45 samples delay of 178 Hz signal at 10 kHz sampling rate) produced the best results and the lowest gain for stabilization. It was necessary to find the linear gain required to stabilize the system to use as a baseline for analyzing pulse width control. It was found that it took less gain to maintain stability than it did to achieve it. The amplitude of the limit cycle is shown as a function of the linear control gain for both increasing and decreasing gain is shown in Figure 4-11, and the numerical data is shown in Table 4-1. Control is achieved at a gain of 1.34 and lost when the gain drops below 1.16. This hysteresis is due to the fact that the poles move differently when approaching the imaginary axis from opposite sides. It can be seen that outside of this hysteresis region the amplitude of the limit cycle is not dependent on whether the gain is increasing or decreasing.

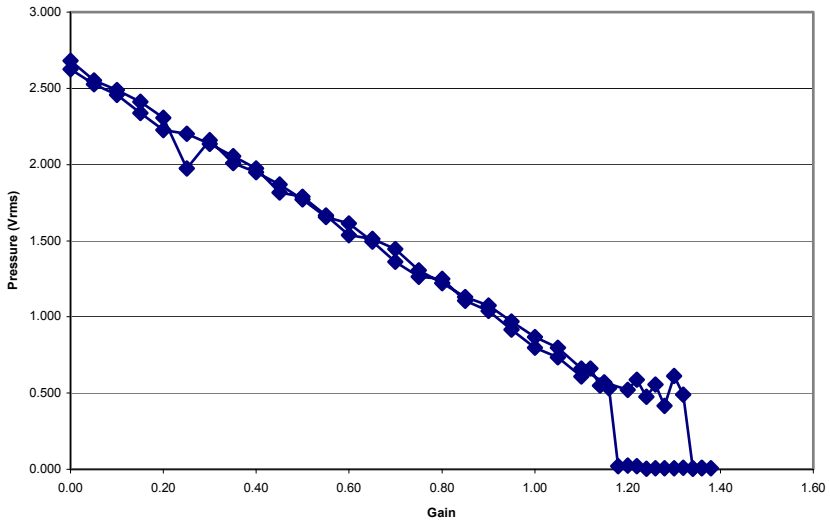


Figure 4-11: Limit cycle amplitude with linear phase shifter

Gain	Limit Cycle Amplitude (V)	Gain	Limit Cycle Amplitude (V)	Gain	Limit Cycle Amplitude (V)	Gain	Limit Cycle Amplitude (V)	Gain	Limit Cycle Amplitude (V)
0.00	2.625	0.70	1.445	1.28	0.415	1.20	0.026	0.65	1.493
0.05	2.527	0.75	1.307	1.30	0.612	1.18	0.020	0.60	1.614
0.10	2.456	0.80	1.223	1.32	0.490	1.16	0.531	0.55	1.656
0.15	2.337	0.85	1.131	1.34	0.008	1.14	0.550	0.50	1.770
0.20	2.225	0.90	1.073	1.36	0.011	1.12	0.663	0.45	1.870
0.25	2.201	0.95	0.968	1.38	0.008	1.10	0.609	0.40	1.950
0.30	2.136	1.00	0.869	1.36	0.008	1.05	0.735	0.35	2.008
0.35	2.056	1.05	0.799	1.34	0.005	1.00	0.799	0.30	2.159
0.40	1.976	1.10	0.662	1.32	0.012	0.95	0.916	0.25	1.975
0.45	1.818	1.15	0.570	1.30	0.007	0.90	1.041	0.20	2.307
0.50	1.789	1.20	0.523	1.28	0.006	0.85	1.107	0.15	2.412
0.55	1.667	1.22	0.589	1.26	0.006	0.80	1.250	0.10	2.489
0.60	1.538	1.24	0.477	1.24	0.004	0.75	1.265	0.05	2.553
0.65	1.514	1.26	0.556	1.22	0.023	0.70	1.362	0.00	2.683

Table 4-1: Limit cycle amplitude vs. gain data

4.4.2 Loss of control at high gains

There are multiple poles associated with the controlled system, including many poles from the input and output filters. If the gain of the linear phase shifter is increased past the gain needed for stabilization, certain poles can cross the imaginary axis into the right half plane and create a secondary instability as explained in Section 3.5. Gains above 2.8 resulted in a power spectrum with a split peak, indicating two potential instabilities evenly spaced around the original instability frequency. Experimentally it was determined that a gain of 4.2 would cause this secondary instability to occur. This peak splitting is because two different poles are going unstable at nearly the same time. The phenomenon can be seen in Figure 4-12 as the gain is increased in increments of 1.5 from 0 to 4.5. Initially the thermoacoustic limit cycle exists. As the gain is increased to 1.5, control is achieved and one stable limit cycle (control-induced) exists. As the gain is further increased to 3.0, peak splitting occurs. The original limit cycle is visible, but has been driven further down and two additional peaks have arisen to either side of it. Finally, at the gain of 4.5, control is again lost. This characteristic has been well documented for phase shift controllers [14]; however, the characteristic is presented in this document to fully characterize the performance of the linear phase shifter's effects on the plant.

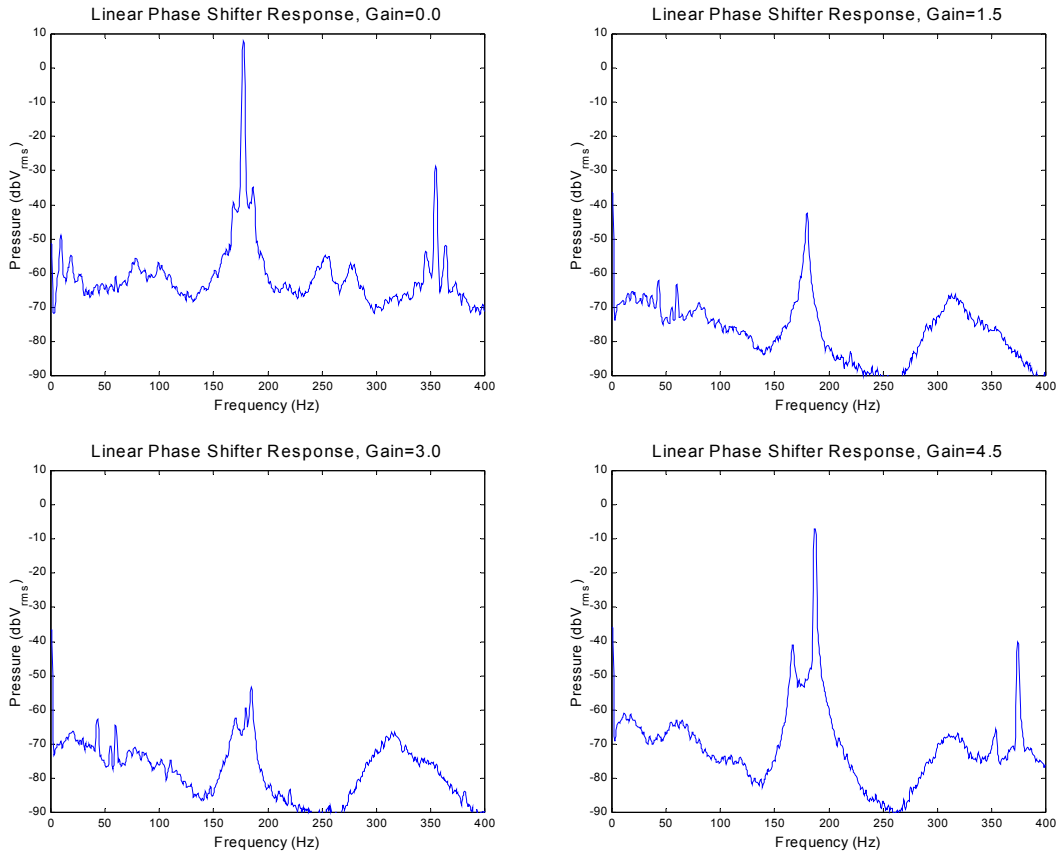


Figure 4-12: Control level with linear phase shifter, various gains

4.5 Validation of the Instability Frequency Hypothesis for Pulsed Control

If the mechanism of control in the tube combustor is indeed the component of the control signal at the instability frequency, we can verify this by examining the ratios of the gains needed to achieve stability for various duty cycle pulse train signals. The Fourier component of the signal at the instability frequency is related to the duty cycle of the signal and is proportional to the Fourier coefficient of the pulse train corresponding to the limit cycling frequency. Thus, for a half harmonic pulse train, the second Fourier component is the sine wave component of the signal at the instability (limit cycling) frequency.

Experiments were done using various duty cycles, and the gains necessary to stabilize the tube combustor were found. The relative gain needed to control the system for a 1/2 signal is shown in Figure 4-13. This is simply the reciprocal of the second

Fourier component. It can be seen here that a 50% duty cycle will require infinite gain to control the system, and is thus incapable of controlling the system. The lowest gain will be needed with a 25% duty cycle signal, which will correspond to a pulse height equal to half the instability period.

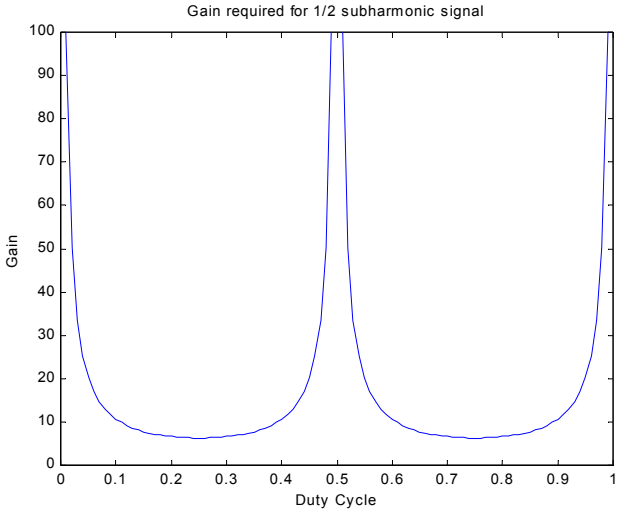


Figure 4-13: Relative gain necessary for various duty cycles

4.5.1 One-half subharmonic control

For the 1/2 signal case, tests were done using a 25.2% duty cycle, 17.2% duty cycle, and 10.0% duty cycle. A 25.2% duty cycle signal should require the lowest gain, and the 17.2% and 10.0% duty cycles correspond to other gains along the curve shown in Figure 4-13 without being on such a steep slope as to be overly sensitive to inaccuracy in the duty cycle. Control tests were done with a linear phase shifter, and a 1/1 signal with duty cycles of 19.6%, 34.6%, and 50.4%. These duty cycles give pulse widths equal to the 1/2 signal cases. Tests were done with a 100° phase shift of the control signal as determined by the HP analyzer, which monitors the total phase of the loop including filters as seen in Figure 4-8. The instability was clearly evident, with no instrumentation, as an audible tone at 178 Hz. The gain was increased until this tone was no longer audible. The gain was then reduced until control was lost and the instability was audible again. The gains required are shown in Table 4-2 for gain of control and Table 4-3 for a loss of control.

It is seen that in all cases achievement of control resulted in a pressure reduction of more than 45 dB compared to an uncontrolled pressure of approximately +8 dB. It should be noted that with the gain set at just enough to achieve control, secondary instabilities were not generated. Further, an increase in gain would not reduce the instability farther, as the input to the DSpace board is at the limits of its A/D converters and thus will not see a smaller signal. Thus it is possible to achieve as much control as the DSP board allows without worrying about secondary instabilities.

Duty Cycle	*# of Hold Samples*	*# of Delay Samples*	*Subharmonic Ratio*	*Gain Required*	*Phase Delay (°)*	*Pressure (dBV_{rms})*
10.0%	11	26	2	3.19	99.4	-42.1
17.2%	19	21	2	2.06	101.5	-41.9
25.2%	28	18	2	1.90	98.2	-44.4
19.6%	11	25	1	1.76	105.0	-43.5
34.6%	19	21	1	1.19	102.0	-45.8
50.4%	28	17	1	1.08	103.3	-42.8
Shifter	-	45	-	1.32	98.3	-43.3

Table 4-2: Gains required to achieve control for various control signals, half-harmonic

Duty Cycle	*# of Hold Samples*	*# of Delay Samples*	*Subharmonic Ratio*	*Gain Required*	*Phase Delay (°)*	*Pressure (dBV_{rms})*
10.0%	11	26	2	2.90	93.9	-9.0
17.2%	19	21	2	1.86	94.7	-8.6
25.2%	28	18	2	1.62	91.6	-8.7
19.6%	11	25	1	1.57	100.0	-6.1
34.6%	19	21	1	1.01	101.6	-6.3
50.4%	28	17	1	0.93	103.3	-6.4
Shifter	-	45	-	1.16	89.6	-4.6

Table 4-3: Gains at loss of control for various control signals, half-harmonic

The ratios between each of the gains required to achieve control was then tabulated in Table 4-4. These ratios are calculated as the gain for the duty cycle in the column divided by the gain of the duty cycle in the row. This ratio will be important below. The same data for a loss of control is shown in Table 4-5.

Actual Ratios								
Subharmonic			Non Subharmonic			Phase		
10%	17%	25%	20%	34%	50%	Shifter		
3.19	2.06	1.90	1.76	1.19	1.08	1.32		
	0.646	0.596	0.552	0.373	0.339	0.414	3.19	10%
		0.922	0.854	0.578	0.524	0.641	2.06	17%
			0.926	0.626	0.568	0.695	1.90	25%
				0.676	0.614	0.750	1.76	20%
					0.908	1.109	1.19	34%
						1.222	1.08	50%
							1.32	P.S.

Table 4-4: Ratios of gains required to achieve control for 1/2 signal, 1/1 signal, and phase shifter

Actual Ratios								
Subharmonic			Non Subharmonic			Phase		
10%	17%	25%	20%	34%	50%	Shifter		
2.90	1.86	1.62	1.57	1.01	0.93	1.16		
	0.641	0.559	0.541	0.348	0.321	0.400	2.90	10%
		0.871	0.844	0.543	0.500	0.624	1.86	17%
			0.969	0.623	0.574	0.716	1.62	25%
				0.643	0.592	0.739	1.57	20%
					0.921	1.149	1.01	34%
						1.247	0.93	50%
							1.16	P.S.

Table 4-5: Ratios of gains for loss of control for 1/2 signal, 1/1 signal, and phase shifter

The Fourier coefficients of the signals used for the experiment were then calculated. The exact duty cycles of the signals were found with an oscilloscope and these duty cycles were used to calculate the coefficients. These coefficients are tabulated similarly to Table 4-6, although the ratios are the coefficient of the row divided by the coefficient of the column. The reason for this inversion will be seen below.

Expected Ratios								
Subharmonic			Non Subharmonic			Phase		
10%	17%	25%	20%	34%	50%	Shifter		
0.3742	0.5617	0.6366	0.7354	1.1271	1.2731	1.0000		
	0.666	0.588	0.509	0.332	0.294	0.374	0.3742	10%
		0.882	0.764	0.498	0.441	0.562	0.5617	17%
			0.866	0.565	0.500	0.637	0.6366	25%
				0.652	0.578	0.735	0.7354	20%
					0.885	1.127	1.1271	34%
						1.273	1.2731	50%
							1.0000	P.S.

Table 4-6: Ratios of Fourier coefficients for signals used experimentally in half harmonic tests

To achieve an equal amount of control, two signals must have the same component at the fundamental frequency. This is seen below, where the signal *s* is the gain of the controller *k* times the Fourier coefficient c_k.

$$s_1 = s_2$$

Substituting for the signal we find

$$(k_1)(c_{k_1}) = (k_2)(c_{k_2})$$

Rearranging we then conclude

$$\frac{k_1}{k_2} = \frac{c_{k_2}}{c_{k_1}}$$

(actual) = (expected)

Thus the ratio of the gains should be equal to the inverse ratio of the Fourier coefficients. An example of this is now done. From Table 4-4 it can be seen that the gain required to achieve control was 1.90 for the 1/2 25% case and 1.76 with a 1/1 20% case. From Table 4-6 it can be seen that the second Fourier coefficient for a 25% signal is 0.6366, and the first Fourier coefficient of a 20% signal is 0.7354. Thus we can compare the ratios and find

$$\frac{k_{1/1,20\%}}{k_{1/2,25\%}} = \frac{1.76}{1.90} \qquad \frac{0.6366}{0.7354} = \frac{c_{2,25\%}}{c_{1,20\%}}$$

$$\therefore 0.926 \cong 0.866$$

These values can also be read directly from Table 4-4 and Table 4-6. In this case, there is a 7.0% error between the observed gain ratio and the expected gain ratio based on the hypothesis. Note that there will be some error due to the hysteresis of the system, since the gains to achieve control and the gains to lose control are both compared to identical Fourier coefficients. The actual ratios of the gain of every signal to the gain of every other signal were computed and are tabulated in Table 4-7 for achievement of control and in Table 4-8 for loss of control. Figure 4-14 shows the error graphically by

plotting the expected ratio vs. the actual ratio. Ideally the points should all lie exactly on the 45° line shown, with the expected ratio equal to the actual ratio.

The subharmonic signals agree very closely when compared to other subharmonic signals. However, there is a slight amount of error between the ratios of subharmonic signals and the fundamental, as well as the subharmonic signals and the phase shifter. This is most likely due to nonlinear effects in the combustor. It seems that the subharmonic component can act to reduce the gain necessary to control the system. This can be also seen from experiments done by Vaudrey [15] where a large amplitude sine wave input to the combustor tube will be spread to many frequencies surrounding the input frequency through the nonlinear dynamics of the system. In general, however, the test results agree closely with the expected results and appear to verify the hypothesis that the limit cycle frequency component of the control signal is responsible for control of the system.

Error							
Subharmonic			***Non Subharmonic***			***Phase***	
10%	***17%***	***25%***	***20%***	***34%***	***50%***	***Shifter***	
-3.1%	1.3%	8.4%	12.4%	15.2%	10.6%		***10%***
	4.5%	11.9%	15.9%	18.8%	14.1%		***17%***
		7.0%	10.9%	13.7%	9.1%		***25%***
			3.6%	6.2%	2.0%		***20%***
				2.5%	-1.6%		***34%***
					-4.0%		***50%***
							P.S.

Table 4-7: Error of gains to achieve control between 1/1, 1/2, and linear phase shifter signals

Error							
Subharmonic			***Non Subharmonic***			***Phase***	
10%	***17%***	***25%***	***20%***	***34%***	***50%***	***Shifter***	
-3.7%	-5.0%	6.4%	4.9%	9.1%	6.9%		***10%***
	-1.3%	10.5%	9.0%	13.3%	11.0%		***17%***
		12.0%	10.4%	14.8%	12.5%		***25%***
			-1.4%	2.6%	0.5%		***20%***
				4.0%	1.9%		***34%***
					-2.0%		***50%***
							P.S.

Table 4-8: Error of gains for loss of control between 1/1, 1/2, and linear phase shifter signals

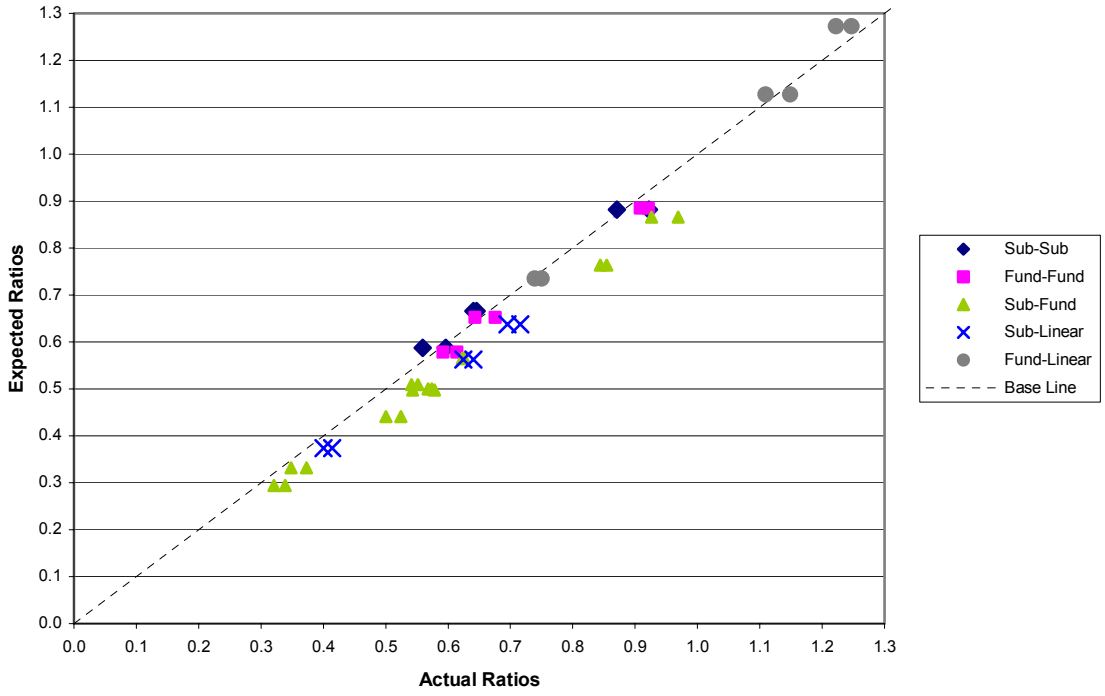


Figure 4-14: Actual vs. expected ratios for half-harmonic signals

4.5.2 One-third subharmonic control

Similar tests were done using a one-third subharmonic control signal. Duty cycles of 7.2%, 12.0%, and 16.6% were tested. These duty cycles have pulse widths approximately equal to those considered in the previous section. As in the half harmonic section, the gains for each of these signals was found. The gains required to achieve control are shown in Table 4-9 and the gains for loss of control in Table 4-10.

Duty Cycle	*# of Hold Samples*	*# of Delay Samples*	*Subharmonic Ratio*	*Gain Required*	*Phase Delay (°)*	*Pressure (dBV_{rms})*
7.2%	12	25	3	3.65	102.2	-39.7
12.0%	20	21	3	2.76	98.2	-42.1
16.6%	28	17	3	2.50	97.4	-43.5
21.8%	12	25	1	1.52	101.9	-42.6
34.6%	19	21	1	1.06	104.6	-43.7
50.0%	28	14	1	0.98	102.8	-42.5
Shifter	-	45	-	1.32	98.3	-43.3

Table 4-9: Gains required to achieve control for various control signals, third-harmonic

Duty Cycle	# of Hold Samples	# of Delay Samples	Subharmonic Ratio	Gain Required	Phase Delay (°)	Pressure (dBV_{rms})
7.2%	12	25	3	3.52	108.8	-7.7
12.0%	20	21	3	2.60	102.4	-8.4
16.6%	28	17	3	2.35	103.6	-10.7
21.8%	12	25	1	1.40	100.6	-6.0
34.6%	19	21	1	0.95	105.5	-5.8
50.0%	28	14	1	0.87	102.5	-6.6
Shifter	-	45	-	1.16	89.6	-4.6

Table 4-10: Gains at loss of control for various control signals, third-harmonic

As above, ratios between every case were calculated and compared to theoretical expectations. The actual ratios for achievement and loss of control are shown in Table 4-11 and Table 4-12. The Fourier coefficients for this case are shown in Table 4-13, and the errors are shown in Table 4-14 and Table 4-15. The actual and expected ratios are plotted in Figure 4-15. As in the previous section, perfect agreement would be when all points lie on the 45° line.

The ratios between gains for different duty cycles using both fundamental frequency control and subharmonic control match closely with what is expected. Ratios between subharmonic and fundamental pulse trains and ratios compared to a linear phase shifter are somewhat farther away from the expected than in the half-harmonic case. Having two subharmonic frequency components present seems to enhance the nonlinear effect noted in 4.5.1, requiring even less gain to control the system. Overall, however, the results are close to expected and appear to verify that the Fourier component at the limit cycling frequency is the dominant means of control.

Actual Ratios								
Subharmonic			Non Subharmonic			Phase		
7%	12%	16%	20%	34%	50%	Shifter		
3.65	2.76	2.50	1.52	1.06	0.98	1.34		
	0.756	0.685	0.416	0.290	0.268	0.367	3.65	10%
		0.906	0.551	0.384	0.355	0.486	2.76	17%
			0.608	0.424	0.392	0.536	2.50	25%
				0.697	0.645	0.882	1.52	20%
					0.925	1.264	1.06	34%
						1.367	0.98	50%
							1.34	P.S.

Table 4-11: Ratios of gains required to achieve control for 1/3 signal, 1/1 signal, and phase shifter

Actual Ratios								
Subharmonic			Non Subharmonic			Phase		
7%	12%	16%	20%	34%	50%	Shifter		
3.52	2.60	2.35	1.40	0.95	0.87	1.16		
	0.739	0.668	0.398	0.270	0.247	0.330	3.52	10%
		0.904	0.538	0.365	0.335	0.446	2.60	17%
			0.596	0.404	0.370	0.494	2.35	25%
				0.679	0.621	0.829	1.40	20%
					0.916	1.221	0.95	34%
						1.333	0.87	50%
							1.16	P.S.

Table 4-12: Ratios of gains for loss of control for 1/3 signal, 1/1 signal, and phase shifter

Expected Ratios								
Subharmonic			Non Subharmonic			Phase		
7%	12%	16%	20%	34%	50%	Shifter		
0.2664	0.3840	0.4244	0.7354	1.1271	1.2731	1.0000		
	0.694	0.628	0.362	0.236	0.209	0.266	0.2664	10%
		0.905	0.522	0.341	0.302	0.384	0.3840	17%
			0.577	0.377	0.333	0.424	0.4244	25%
				0.652	0.578	0.735	0.7354	20%
					0.885	1.127	1.1271	34%
						1.273	1.2731	50%
							1.0000	P.S.

Table 4-13: Ratios of Fourier coefficients for signals used experimentally in third harmonic tests

Error							
Subharmonic			Non Subharmonic			Phase	
10%	17%	25%	20%	34%	50%	Shifter	
9.0%	9.1%	15.0%	22.9%	28.3%	37.8%		10%
	0.1%	5.5%	12.7%	17.7%	26.4%		17%
		5.4%	12.6%	17.6%	26.3%		25%
			6.9%	11.6%	19.9%		20%
				4.4%	12.2%		34%
					7.4%		50%
							P.S.

Table 4-14: Error of gains to achieve control between 1/1, 1/3, and linear phase shifter signals

Error						
Subharmonic			Non Subharmonic			Phase
10%	17%	25%	20%	34%	50%	Shifter
6.5%	6.4%	9.8%	14.2%	18.1%	23.7%	10%
	-0.1%	3.1%	7.2%	10.9%	16.2%	17%
		3.2%	7.4%	11.1%	16.3%	25%
			4.0%	7.6%	12.7%	20%
				3.4%	8.3%	34%
					4.7%	50%
						P.S.

Table 4-15: Error of gains for loss of control between 1/1, 1/3, and linear phase shifter signals

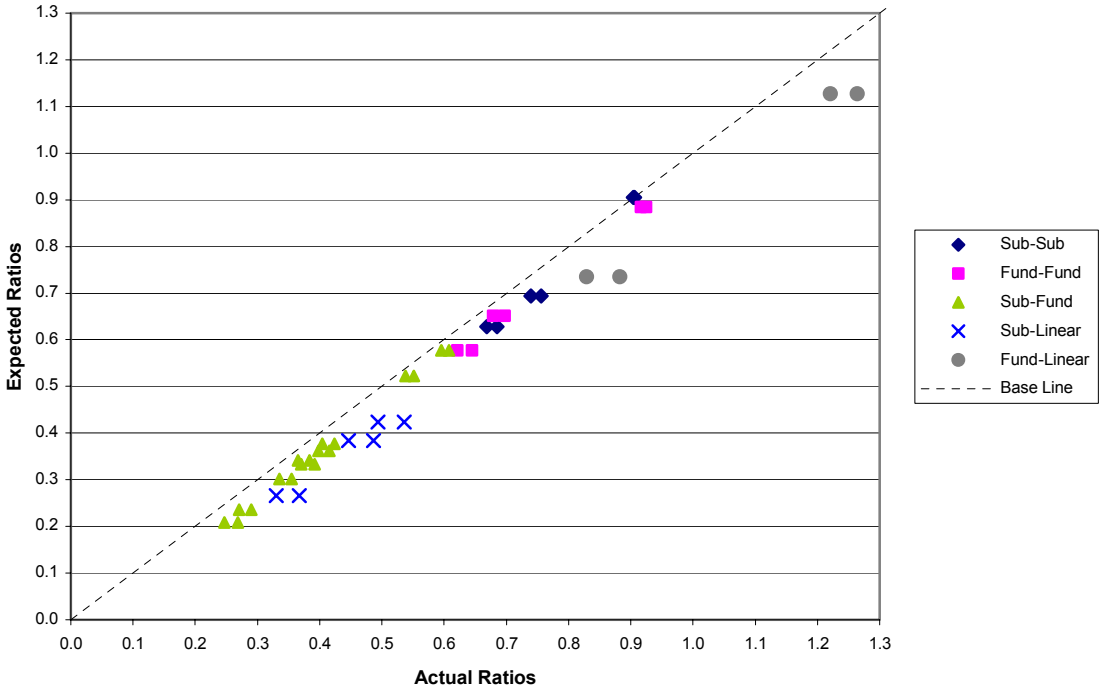


Figure 4-15: Actual vs. expected ratios for third harmonic signals

4.6 Validation of Fixed Pulse Height Analysis

According to the theory shown in Section 3.3, the pulse height required to stabilize the system should be easily calculated from the describing function (repeated below, ignoring phase) and the limit cycle amplitude curve from a linear phase shifter as shown in Figure 4-11.

$$DF = \frac{2X}{AM\pi}\left|\sin(\frac{w}{T_0}\pi)\right|$$

When a 1/1 50% signal is used (M=1, $\frac{w}{T_0}$=0.5) the describing function can be further simplified to

$$DF = \frac{2X}{A\pi}$$

Therefore, for a fixed height X the describing function can be plotted for a range of pressures A. When the describing functions for various pulse heights are plotted and superimposed on the hysteresis curve as shown in Figure 4-16, the number of intersections for a particular pulse height can be clearly seen. As can be seen in the plot, pulse heights above 0.8 V should result in control of the system, as there are no intersections producing limit cycles for pulse heights above 0.8 V.

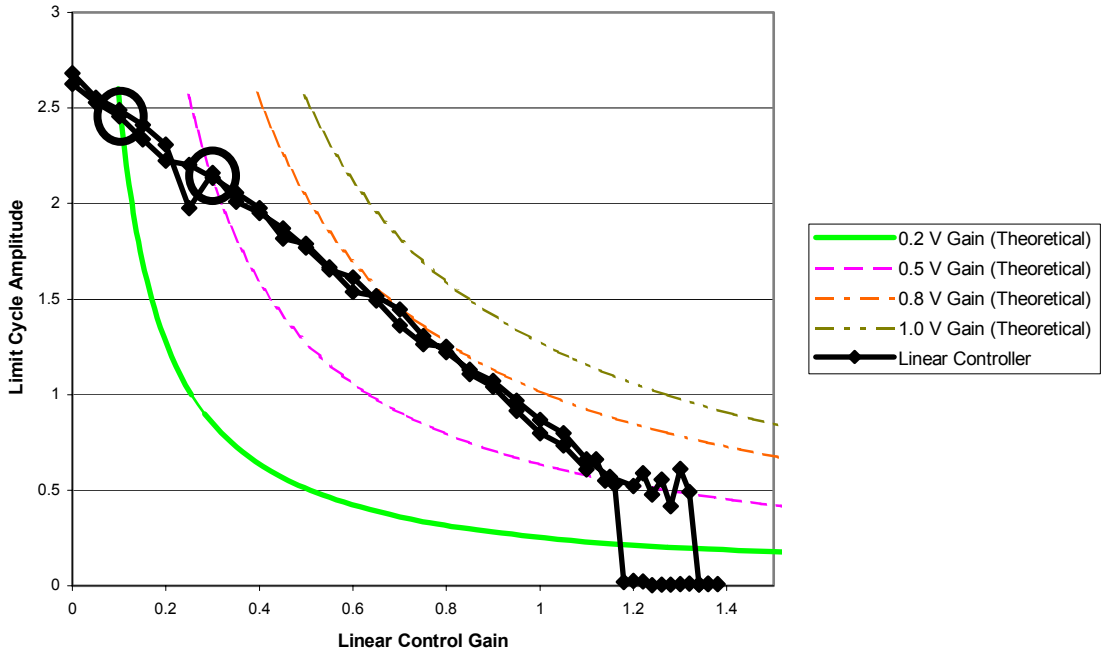


Figure 4-16: Describing functions gains for various pulse heights superimposed on limit cycle amplitude curve and intersections shown

Experiments were then run with various pulse heights. All signals were 1/1 50%. The intersections for pulse heights below 0.8 V are circled on Figure 4-16. The level of control was examined for several different pulse heights, and these values are tabulated in

Table 4-16. The level of control is shown graphically in Figure 4-17. These are the thermoacoustic limit cycles that will result when the particular control signal is applied. It is seen that the level of control can be different for equal pulse heights, in this case depending on whether the height is increasing or decreasing. This is due to the system starting on opposite sides of the unstable limit cycle. In addition, the frequency of the instability for pulse heights above 0.6 V shifts due to the ultimate amplitude being reached, as explained in Section 3.5. This causes secondary instabilities that will occur at frequencies determined by the filter pole frequencies. As such the lower curved part of the plot can be recognized as a line of constant gain of 4.2. As the pulse height decreases, the level of the instability will decrease accordingly.

Using any control signal where both intersections exist (thermoacoustic limit cycle and control-induced via ultimate gain), it was fairly trivial to move between the thermoacoustic and control-induced limit cycles. This will be at any pulse height below 0.6 V in Figure 4-17. A disturbance created by interrupting the airflow at the top of the combustor tube was enough to cause the combustor to jump from one instability to the other, with no change to the controller. In essence the disturbance was enough to cause the system to be able to jump past the unstable limit cycle. The farther apart the two limit cycles, the greater the disturbance needed to cause this jump. As the pulse height increased and the two limit cycles became closer in amplitude, it was easy enough to jump between the two poles that the jump would occur spontaneously, with no applied disturbance. In these cases the control-induced limit cycle, or the lower line on the plot shown, was the "dominant" limit cycle. Indeed, for pulse heights between 0.6 V and 0.8 V an applied disturbance would cause the system to temporarily jump to a thermoacoustic limit cycle and then immediately return to the control-induced limit cycle. This explains the discrepancy between the break-off point at 0.6 V as shown in Figure 4-17 and the expected break-off point of 0.8 V shown in Figure 4-16.

Pulse Height	*Pressure (Power Spectrum)*	*Pressure (Linear Spectrum)*	*Instability Frequency*
(V)	*(dB)*	*(V)*	*(Hz)*
0	8.205	2.533	177
0.1	8.047	2.56	177
0.2	7.661	2.45	177
0.3	7.31	2.355	177
0.4	6.897	2.301	178
0.5	6.278	2.142	178
0.6	-15.77	0.1789	187
0.7	-13.269	0.2118	187
0.8	-12.062	0.253	187
0.9	-11.181	0.2608	187
1	-10.455	0.2788	187
0.9	-11.255	0.2649	187
0.8	-12.199	0.2603	187
0.7	-13.126	0.2189	187
0.6	-14.605	0.1851	187
0.5	-16.264	0.1593	187
0.4	-18.351	0.1142	187
0.3	-21.388	0.0868	187
0.2	-25.155	0.0537	187
0.1	-32.085	0.0243	188
0.08	-33.745	0.022	188
0.06	-36.273	0.0154	166
0.04	-40.705	0.00527	167
0.02	-50.955	0.00317	165
0.01	8.665	2.6	178

Table 4-16: Level of control for fixed pulse height signals

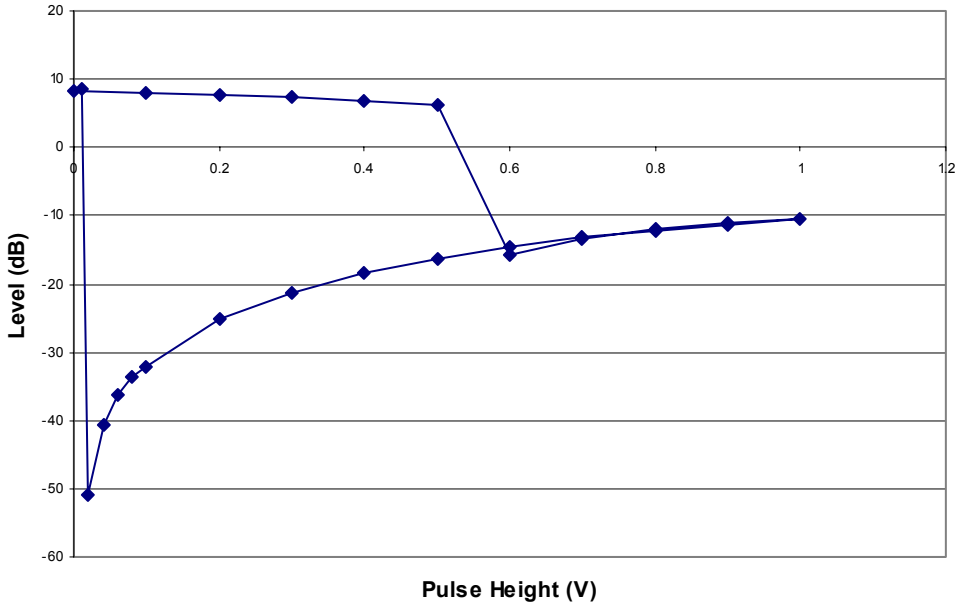


Figure 4-17: Instability level for fixed pulse height controller

Further verification of the validity of the theory is possible by comparing the experimental limit cycle heights with those predicted in Figure 4-16. The actual value of the limit cycle with a few different control signals is shown as a line in Figure 4-18, and it can be clearly seen that the actual level is at the intersection of the describing function and the linear phase shifter results.

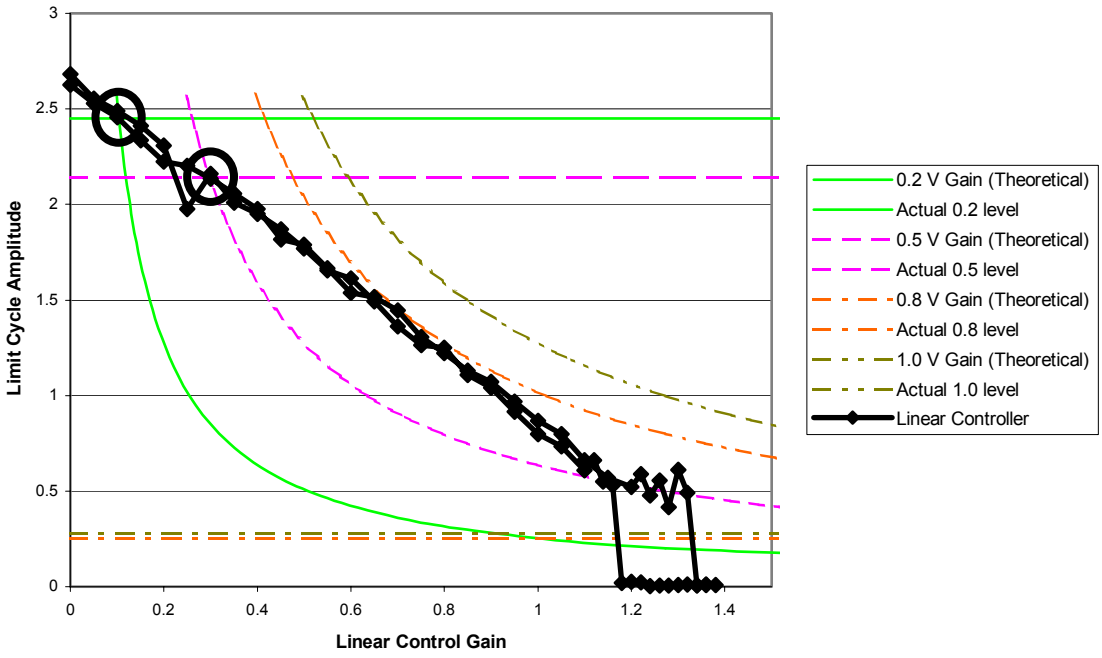


Figure 4-18: Actual levels of limit cycle shown with intersections circled

Further evidence can also be gathered by examining the control-induced limit cycles. As explained in Section 3.5, the control-induced stable limit cycle should correspond to a gain of k_{ult}, which was shown to be approximately 4.2 for this system (section 4.4.2). Examining the control-induced limit cycle with a pulse height of 0.9 V, for instance, it is seen that the amplitude of the limit cycle is 0.276 V. Note that this is a different value than shown in

Table 4-16, since this value the amplitude of the input signal immediately before entry into the controller. Since the filters may change the amplitude, the level will be different from the unfiltered signal. Not also that the unfiltered signal was recorded in V_{rms} instead of volts. The gain can then be found to be

$$\frac{4(0.9)}{0.276\pi} = 4.15$$

The gain at the control-induced limit cycle for all pulse heights is shown in Figure 4-19. It can be seen that the gain tracks very closely with the expected value of 4.2. The inaccuracy at the 0.08 V point is likely due to the extremely small amplitude (9 mV) of

the instability and the possibility of noise at this low level. Therefore the ultimate gain of the system is indeed limiting the amount of reduction that can be achieved by the controller.

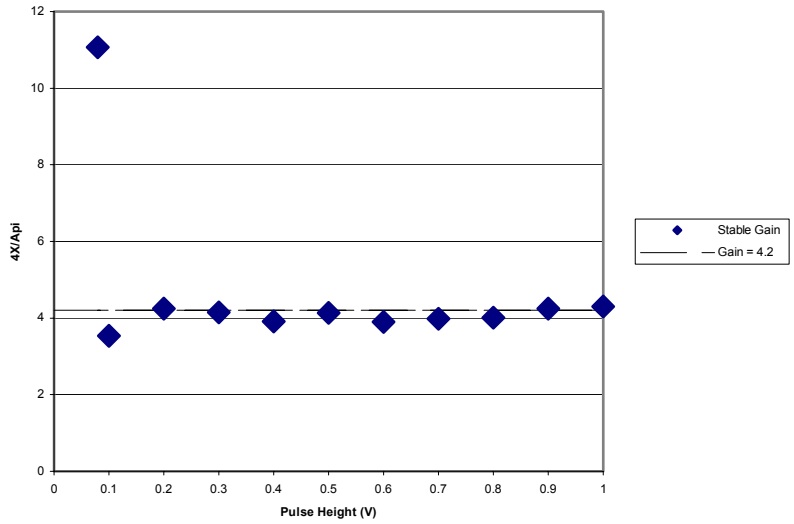


Figure 4-19: Gain of fixed pulse height controller

4.6.1 Variable subharmonic ratio results

Although extensive testing and fine-tuning has yet to be done, some simple experiments were completed using *variable subharmonic ratios* for control. Variable subharmonic ratios means that as the controller achieves control, it increases the value of *M*. This will decrease the gain of the system and allow fewer actuator cycles and possibly lower fuel use. The gain of the system was calculated as the maximum height of the output over the maximum height of the input, and the subharmonic ratio was changed continuously to try to keep this gain near a prescribed value. For now the subharmonic ratio was only changed between 1 and 2. Better ratio switching techniques in the future will cause a further reduction in the number of cycles and an increase in control of the system. As seen in Figure 4-20, as the pressure trace begins to grow, the subharmonic ratio returns to 1 to halt the growth of the signal. When it is again stabilized, it will return to a 1/2 subharmonic square wave to reduce the number of cycles.

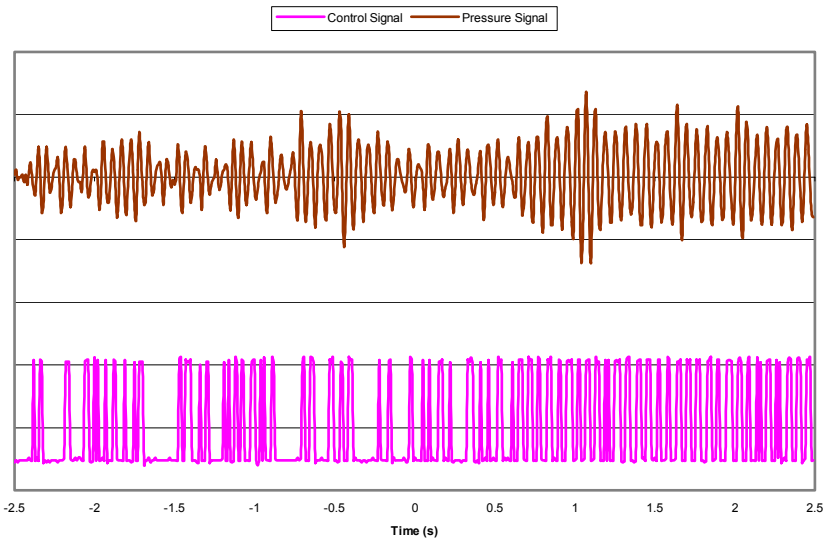


Figure 4-20: Results with variable subharmonic ratio forcing

4.6.2 Variable duty cycle fixed height results

Experiments were also done with variable duty cycle, fixed height control signals, using a subharmonic ratio of one. Note from the equation

$$C_M = \frac{X}{M\pi}\sin(\frac{w}{T}\pi)$$

that the Fourier coefficient of the control signal is periodic, with the maximum value occurring at a duty cycle of 50%. This means that the level of control with a fixed height controller should be maximized with a duty cycle of 50%, and fall off symmetrically with changes in duty cycle away from 50%. Thus a 40% duty cycle signal and a 60% duty cycle signal should have exactly the same results, since the Fourier component at the instability frequency is exactly the same in both cases.

Experiments were done to verify this. A fixed height signal was input to the system at 40%, 50%, and 60% duty cycles. The phase shift of the controller was varied until the frequency of the instability was 186 Hz in each case. This was necessary because the filter introduces a steep phase curve, and any change in phase shift causes a frequency shift along the phase curve to keep a constant loop phase as explained in Section 2.2.2. Therefore, changing the controller phase will not change the phase of the frequency response, but the frequency of the instability. As can be seen in Table 4-17,

the assumption that the control amplitude is maximized at 50% and that the level is identical at 40% and 60% is verified. The level of suppression goes down at 50%. This is at first counterintuitive, as it seems that the 50% signal must achieve less gain. However, the fixed pulse height controller is operating at the ultimate gain of the system. Therefore greater pressure amplitude corresponds to a greater control signal level. It can be seen that since the level of control for both the 40% and 60% duty cycles are identical, the level of the control signal for both cases must be identical.

Duty Cycle	*Pressure (dbVrms)*	*Phase Shift (°)*	*Frequency (Hz)*
40%	-6.98	65.7	186
50%	-5.83	65.7	186
60%	-6.96	65.5	186

Table 4-17: Level of control with variable duty cycles and fixed pulse height

4.7 Open Loop Control

As others in the field have had some success with using open loop pulsed signals, or pacemakers, some experiments were done to examine the phenomena. It was desired to see if the same mechanism of control was causing the success seen by others. It is theorized that the addition of an open loop pressure signal can improve the level of control by aiding in the transfer of heat away from the flame and reducing the amount that is fed into the acoustics. This idea is reinforced by the experiments below. The pacemaker signal was used by itself and then in conjunction with a linear phase shifter.

4.7.1 Pacemaker signal

Initially an open loop signal was used with no other method of control. No filters were used on the output of the signal since phase shift was not important and a pure sine wave was used. The instability before control was at a peak value of 8.3 dBV_{rms} at 178 Hz. The closest the frequency generator (the HP analyzer) could get to an exact half-harmonic signal was 89.3 Hz, which was used along with other signals at various frequencies to test the effects. The pacemaker amplitude was varied from 500 mV to 1250 mV, and the results are shown in Table 4-18. The pressure level seen in the system by the HP analyzer at the pacemaker frequency was recorded as the pacemaker level.

Frequency (Hz)	*75*		*89.3*	
Amplitude (mV)	*Pacemaker Level (dB)*	*Limit Cycle Level (dB)*	*Pacemaker Level (dB)*	*Limit Cycle Level (dB)*
500	-15.37	7.09	-17.88	2.49
750	-11.08	3.99	-14.75	1.74
1000	-8.14	-58.78	-12.37	-12.64
1250	no flame		-10.67	-24.74

Frequency (Hz)	*95*		*110*	
Amplitude (mV)	*Pacemaker Level (dB)*	*Limit Cycle Level (dB)*	*Pacemaker Level (dB)*	*Limit Cycle Level (dB)*
500	-19.98	3.68	-22.11	3.98
750	-16.79	3.39	-18.72	4.91
1000	-14.45	-1.88	-16.28	2.86
1250	-12.67	-54.88	-14.45	-31.05

Table 4-18: Open loop pacemaker results

As seen above, some level of control may be gained by using an open loop signal. However, the characteristics of the flame had changed. The appearance of a normally controlled flame is steady and flat. In the cases above where a large amount of control was realized, it was seen that the flame was not steady and appeared cupped. It also seemed to jump around a lot. In fact, with a 1250 mV 75 Hz pacemaker signal, the flame was extinguished due to the large amount of pressure injected into the combustor. In addition, for some large amplitude pacemaker signals, a low-level pressure component could be seen at double the pacemaker frequency. This was seen as a 190 Hz signal in the case of a 95 Hz pacemaker, and a 150 Hz signal in the case of a 75 Hz pacemaker. This simply demonstrates the nonlinear nature of the combustion process. It also illustrates how some of the energy at a subharmonic frequency may be converted to energy at the instability frequency and thus contributes to control. This was seen in the proportional controller of 4.5.1 and 4.5.2, where using a subharmonic control signal required slightly less (~15%) gain than was predicted by Fourier analysis alone. This conclusion led to further experiments with pacemakers using phase shifters.

4.7.2 Pacemaker signal with phase shift controller

A lower frequency pacemaker signal was then combined with a linear phase shift controller. The phase shifter was set to a non-optimal phase shift setting so as to allow an examination of the effect of the pacemaker. For this case, the uncontrolled instability

was 6.55 dBV_{rms} at 177 Hz. The phase shift controller only reduced the level at 177 Hz to –6.98 dBV_{rms}, a reduction of about 14 dB. Pacemaker signals at three different frequencies were then added, and the results are shown in Table 4-19. Once again the pacemaker level is the level as read by the pressure transducer measuring the limit cycle.

Frequency (Hz)	***75***		***89.3***		***100***	
Amplitude (mV)	***Pacemaker Level (dB)***	***Limit Cycle Level (dB)***	***Pacemaker Level (dB)***	***Limit Cycle Level (dB)***	***Pacemaker Level (dB)***	***Limit Cycle Level (dB)***
500	-19.10	-5.10	-19.63	-7.57	-21.65	-10.93
750	-15.12	-30.53	-16.33	-29.12	-18.44	-31.02
1000	-12.21	-26.32	-14.07	-28.75	-16.16	-53.30
1250	-10.32	-47.52	-12.37	-29.11	-14.49	-59.07

Table 4-19: Control levels for phase shifter with pacemaker signal added

It can be seen here that the pacemaker signal is effective at adding control authority to the system. It does appear to be dependent on the frequency of the pacemaker. At a half-harmonic frequency, the pressure suppression seemed to be at a maximum nearly immediately, with the limit cycle level staying constant at approximately –29 dB. At other frequencies more control was established at higher pacemaker levels. However, the pacemaker in general induced a large pressure spike that could do as much damage to the combustor as an instability. The flame was also identical to the case above, and did not seem stable. The pacemaker again appears to work by removing heat from the system and breaking the self-excited loop of heat release and acoustic pressure.

4.8 Filtered Subharmonic Control

Since it was theorized that only the component of a subharmonic signal at the instability frequency had any impact on control, it was decided to further test this theory by using the same subharmonic controller as before with a signal that did not contain any component at the instability frequency. For this purpose, a 1/2 subharmonic signal with a duty cycle of 50% was chosen. Note that for this pulse train signal, the second Fourier coefficient will be zero. To further ensure that only the subharmonic was inserted into the system, the filter on the output was changed from 185 Hz to 89 Hz. This eliminated

any component at the instability frequency. Note that this also changed the ideal phase shift necessary to achieve control.

4.8.1 Proportional subharmonic

To begin with, a proportional subharmonic signal was used. Three different gain values were chosen, and the level of control was examined for two different phase shifts. The uncontrolled system had a thermoacoustic instability of 6.59 dBV_{rms} at 177.5 Hz. The results are shown in Table 4-20.

Gain	*0.5*		*1.0*	
Phase Delay	*Subharmonic Level (dB)*	*Limit Cycle Amplitude (dB)*	*Subharmonic Level (dB)*	*Limit Cycle Amplitude (dB)*
0 samples	-12.98	1.23	-12.63	-4.32
29 samples	-14.34	-0.45	-13.44	-6.88

Gain	*1.5*	
Phase Delay	*Subharmonic Level (dB)*	*Limit Cycle Amplitude (dB)*
0 samples	-13.96	-10.48
29 samples	-19.76	-13.36

Table 4-20: Control level with filtered proportional subharmonic control signal

It can be seen here that the controller did not achieve much attenuation of the instability. It should be noted here that all data was taken with 30 averages of the HP analyzer. When the controller was set to a gain of 1.0, the pressure in the system would fluctuate with approximately a 7 second period. At some points there would be more control, and at some points there would be no control. Increasing the gain to 1.5 increased the period to 8 seconds. This is obviously a different form of control than the linear phase shifter, and although the average of the limit cycle may show a reduction, in general this is not a useful form of control. To further reinforce the idea that the control signal must have a component at the instability frequency, this test was repeated for fixed height subharmonic pulses.

4.8.2 Fixed height subharmonic

The above experiment was also run with fixed pulse height signals. The results are shown in Table 4-21. The same reference instability of 6.59 dBV_{rms} at 177.5 Hz was used.

Height (V)	*0.4*		*0.6*	
Phase Delay	*Subharmonic Level (dB)*	*Limit Cycle Amplitude (dB)*	*Subharmonic Level (dB)*	*Limit Cycle Amplitude (dB)*
0 samples	-20.5	-6.52	-16.9	5.45
29 samples	-19.45	2.9	-16.45	4.09

Height (V)	*0.8*		*1.0*	
Phase Delay	*Subharmonic Level (dB)*	*Limit Cycle Amplitude (dB)*	*Subharmonic Level (dB)*	*Limit Cycle Amplitude (dB)*
0 samples	-14.46	3.42	-12.6	-2.29
29 samples	-14.35	2.07	see text	

Table 4-21: Control level with filtered fixed height subharmonic control signal

As above, this controller did not achieve a great reduction in control. However, for the 1.0 V 29 sample delay signal, a reduction to –51.88 dB was seen. This condition caused the subharmonic signal to be the dominant signal in the system. Thus, the subharmonic controller, which watches for zero crossings, now saw the subharmonic control signal as dominant. At this point, the frequency was cut in half again. This cycle was repeated until there were many subharmonics of the instability present. Therefore this controller does not make a valid example. However, it does go to show that the subharmonics of a controller can have an impact on the control of the system. These subharmonics will not reduce the instability with as much efficiency or as consistently as a normal pulsed controller however.

4.9 Liquid Fuel Experiments

Preliminary tests were also done on a liquid fuel combustor using pulsed fuel injection control. Most real combustion rigs will use fuel injection control, so it was desirable to see if the theories verified with acoustic control were valid for fuel injection control as well. This particular fuel rig used primary fuel injection for control.

4.9.1 Experimental setup

The combustor burned ethanol and used swirl air to induce an instability. Lagimoniere 2001 [7] discusses the combustor rig and fuel injection control design in detail. Swirl airflow of 0.0090 scfm and axial airflow of 0.0025 scfm was used, with a fuel flow rate of approximately 0.42 cm^3/s. The fuel was pressurized to 160 psi, and a check valve with a 60 psi pressure drop was installed in the system, giving a fuel pressure of 100 psi across the atomizer. A piston in the fuel line just upstream of the atomizer modulated the fuel pressure. For this particular experiment, the check valve was left open to ensure that the average fuel flow did not change for a controlled system relative to the uncontrolled system. The uncontrolled combustor pressure spectrum in Figure 4-21 shows the instability of approximately 7 dB at just above 100 Hz.

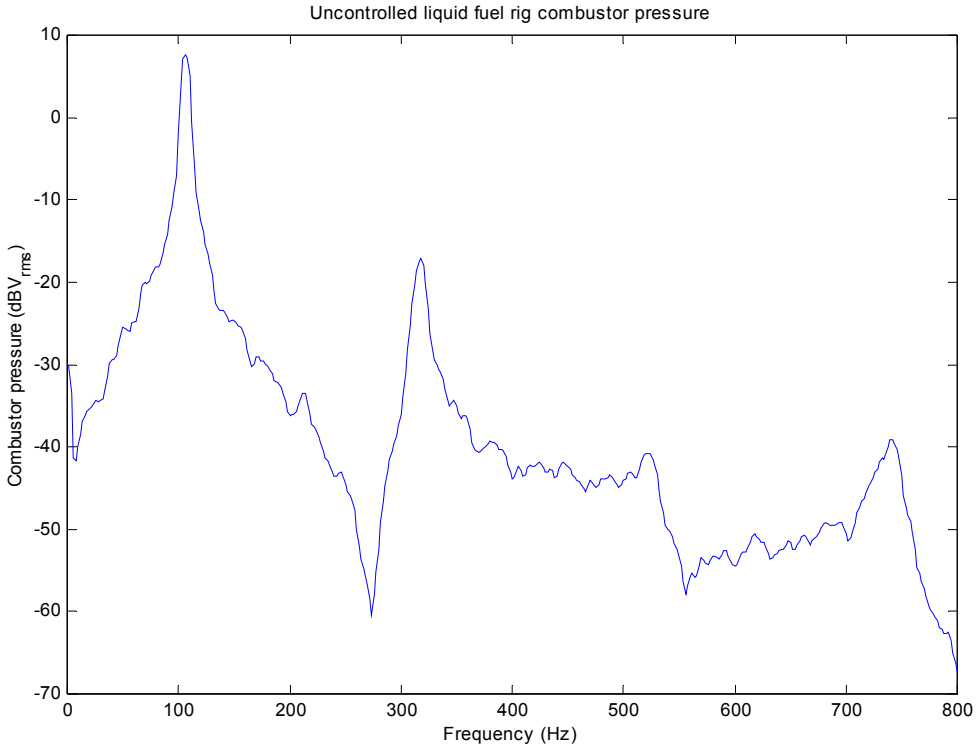


Figure 4-21: Uncontrolled liquid fuel rig combustor pressure

To control the system, pulses were sent to the piston, which was controlled by a piezo stack allowing a displacement of 150 μm. A 1000 volt amplifier drives the piezo, with a 10 V input signal resulting in a 1000 V output. In theory, the integral of the pulsed

control signal should cause the fuel pressure to follow the desired pulsed signal since the signal being sent to the piezo is its position. However, it was found that the fuel pressure in the supply line as measured by a pressure sensor just downstream of the piston followed the control signal nearly exactly. It is theorized that air in the system, or possibly the nitrogen pressurizing the fuel, caused the fuel to act as a compressible liquid. Thus the pressure will vary with the position of the piston. It was also found that when pulsing the fuel, the average fuel pressure in the system stayed at approximately 100 psi. The piston could increase pressure to approximately 180 psi, and during the piston backstroke decrease fuel pressure to 70-80 psi as it refilled the piston chamber.

A 75 Hz open loop, 1/1 30% signal was input into the piston to examine the effect on fuel pressure, and the results are shown in Figure 4-22. The unmodulated fuel pressure is shown as a dashed line, and the modulated fuel pressure is superimposed. The control signal (low pass filtered at 500 Hz) is shown as a dashed line, and has been shifted down 5 V for clarity.

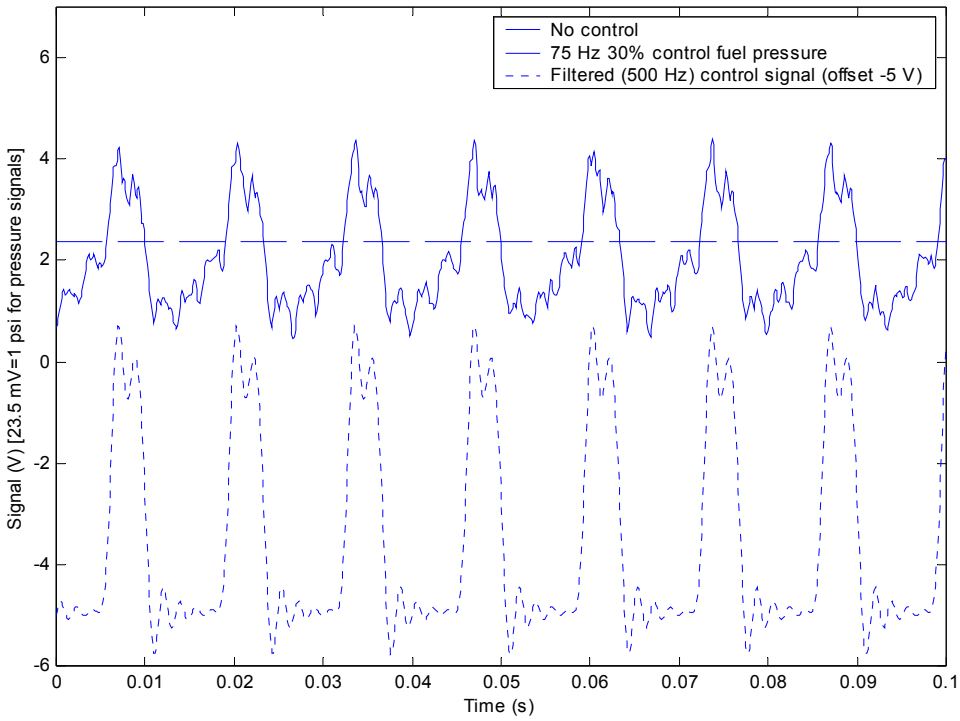


Figure 4-22: Fuel pressure with 75 Hz 1/1 30% open loop signal applied

The effect on combustor pressure for this same open loop signal was then examined, and the pressure is shown in Figure 4-23. The 75 Hz component is clearly seen in the combustor pressure, although it is not as dominant as the limit cycle. In addition, the components of the fuel pressure itself are shown, and the Fourier components at multiples of 75 Hz are clearly visible. Thus the control system will be able to assert authority on the combustor pressure.

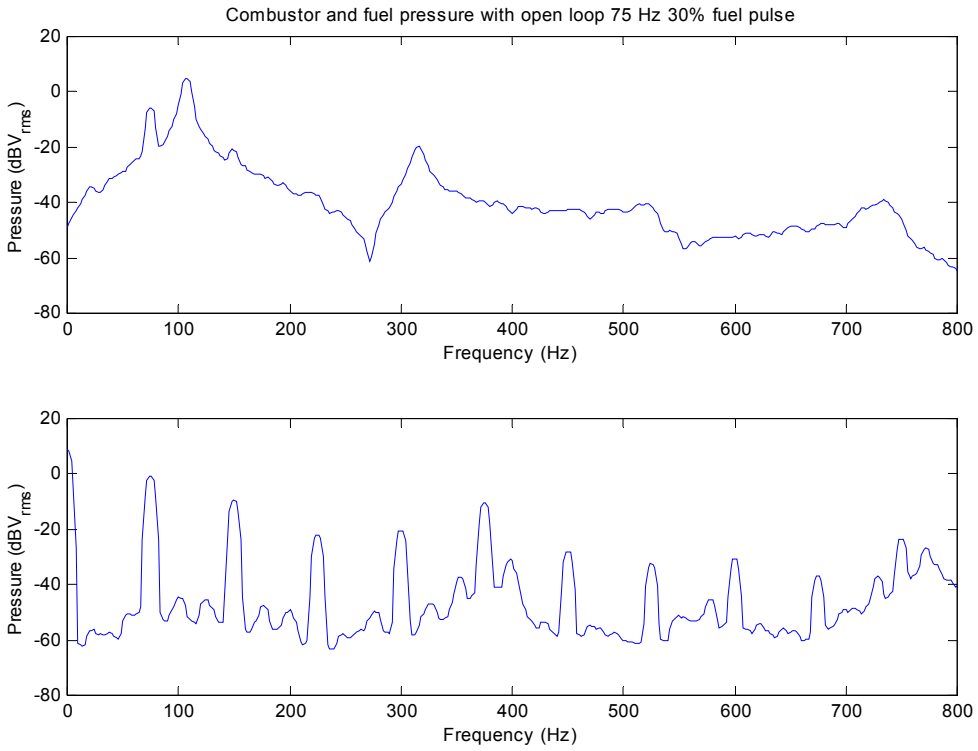


Figure 4-23: Open loop 75 Hz 1/1 30% signal and effect on combustor pressure

Since a piezo can not respond to negative voltages, and the integral signal did not appear to work, the piston output signal was modified slightly from that seen in Appendix A. The same output as used in the acoustic section was used, except that the low value of the pulse was always zero instead of a negative value.

In addition, since the frequency of the instability was approximately 110 Hz, the filter break points were changed. The input filter was simply a low pass filter at 125 Hz, and the output filter was a low pass filter at 500 Hz. The superharmonics were left in place in the output to better simulate a pulsed fuel system. However, the filter was set at 500 Hz since the amplifier driving the piezo only responded to approximately 700 Hz.

4.9.2 Proportional fuel control

Since a piston was used to drive the fuel injection system, it was possible to change the height of the fuel pulses sent to the system. Thus a similar experiment to the ones described in Section 4.5 was run. However, since a linear phase shifter is not valid, all tests were done with pulsed fuel only. First, using a 1/1 50% signal, the ideal phase shift for the controller was found by changing the phase shift with a fixed gain and finding the point where the instability was reduced the most. The uncontrolled system had an instability of 7.25 dBV_{rms} at 112 Hz, and it was found that the best phase between the instability signal and the control signal occurred at –40° of phase shift as read by the HP analyzer. It should be noted that the HP analyzer only reads phase from –180° to +180°, so this corresponds to a phase shift of 320°.

Since it was not possible with the fuel injector to completely eliminate the instability, a set reduction of ~5 dB was selected. Three different duty cycle 1/1 signals were then used (with the phase shifts adjusted so each was at –40°) and the gains required to achieve the 5 dB gain were recorded. This data is shown in Table 4-22.

Duty Cycle	# of Hold Samples	# of Delay Samples	Subharmonic Ratio	Gain Required	Phase Delay (°)	Pressure (dBV_{rms})
20%	19	14	1	2.60	-42.4	2.64
37%	34	8	1	1.54	-40.0	2.44
50%	46	1	1	1.30	-41.1	2.41

Table 4-22: Gains required for control with proportional fuel injection

The ratios between the gains necessary for control were then compared to the ratios between the Fourier coefficients, and the results are shown in Table 4-23. It was not possible to achieve the desired level of control with a half harmonic signal, and thus the exclusion of any data here. When a gain of approximately 2.4 was reached with a half harmonic signal, the output of the controller would be above 10 V. This is above the allowable input to the amplifier, and thus the amplifier will not be able to follow the control signal. A time trace of the data for a 1/2 18% signal with a gain of 2.4 is shown in Figure 4-24. The output can clearly be seen to be above 10 V for several peaks, and thus the resulting clipping. The controller could not reduce the peak to below approximately 4.7 dB with a half harmonic signal. An actuator with a greater stroke would be necessary to achieve control with a half harmonic signal.

Actual Ratios				
Duty Cycle				
20%	37%	50%		
2.60	1.54	1.30		
	0.592	0.500	2.60	20%
		0.845	1.54	37%
			1.30	50%

Expected Ratios				
Duty Cycle				
20%	37%	50%		
0.7484	1.1685	1.2732		
	0.640	0.588	0.7484	20%
		0.918	1.1685	37%
			1.2732	50%

Error		
Non Subharmonic		
20%	37%	50%
-7.6%	-14.9%	
	-8.0%	

20%
37%
50%

Table 4-23: Ratios of gains required to achieve control for proportional fuel injection

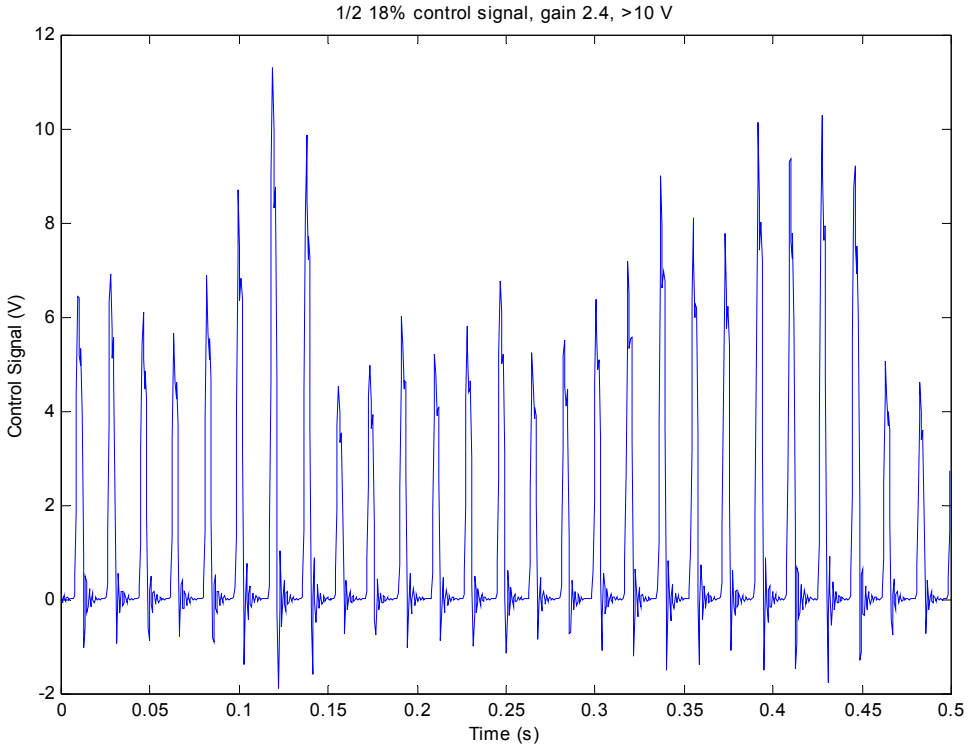


Figure 4-24: 1/2 18% signal, gain=2.4, with output greater than 10 V

It can be seen that in general the errors are reasonably small. This would seem to suggest that the same mechanism of control exists for fuel injection control as for acoustic control. The instability signal is in general much less stable in both frequency

and amplitude as compared to the Rijke tube instability. This caused some errors in the measurements. First, the duty cycle is found by examining the duty cycle calculated by the controller. This duty cycle depends on the instability period, which was not constant in this case. Thus the duty cycle was constantly changing, and the duty cycles shown are only approximately the average. This error will cause some error in the Fourier coefficient calculation.

The experiment was fairly repeatable however. On a subsequent day identical control signals were input into the combustor at 20%, 37%, and 50%, and the resultant instability levels compared. The new levels were at 2.39 dB, 1.95 dB, and 2.92 dB, respectively. These compare well with the levels of the previous day of 2.64 dB, 2.44 dB, and 2.41 dB. In addition, the uncontrolled level was nearly identical, with a peak of 7.12 dB as compared to 7.25 dB the previous day.

With the errors inherent in this process, less than 15% error from the theoretical values is convincing. This is a good indication that the pulsed fuel injection system operates under the same control theory as that examined with a pulsed acoustic system.

4.9.3 Fixed height fuel control

It was also desired to duplicate experiments done with fixed pulse height signals using acoustic control signals. However, due to the lack of a limit cycle amplitude curve as determined by a linear phase shifter, these measurements would have been inconclusive. Thus it was decided to test the hypothesis by attempting to find the ultimate gain of the system and show that the ratios of control heights to instability heights were constant. However, it was found that the system lacked control authority to reach ultimate gain. The maximum stroke of the actuator is 150 μm, and this was reached at a gain of slightly greater than 2. Therefore the ultimate gain could not be reached. In addition, the lack of stability in instability amplitude caused a large amount of measurement error, with the height of the peak changing from 300 mV to 900 mV simply by averaging another data set. Thus it was not possible to reach any significant conclusions with fixed pulse height control signals.

5 Conclusions and Future Work

This work has drawn some important conclusions about the mechanism of control of pulsed control systems. It also provides a good basis for future work on the subject, and suggests some ideas to be explored and expanded on.

5.1 Conclusions

The main conclusion from this work is that combustion instabilities should be eliminated with linear controllers. Since the system exits the nonlinear regime as soon as the instability level is reduced below the limit cycle amplitude, only linear control can cause the poles to move to the left half plane and eliminate the instability completely. Since linear control is necessary, and superposition holds for linear systems, the conclusion is drawn that signals at the instability frequency will have the dominant effect on reducing the instability.

Other researchers in the field have used pulsed and subharmonic control successfully. Based on the above conclusions, it was a curious result that subharmonic signals were effective. However, no detailed mathematical work had been done to investigate the mechanism of this reduction. Therefore this work performed Fourier analysis on subharmonic signals to provide a mathematical foundation. It was found that the Fourier analysis is identical for subharmonic and non-subharmonic pulsed signals. Fourier analysis shows that unless the coefficient of the control signal at the instability frequency (i.e. 2^{nd} coefficient of 1/2 harmonic) is zero, there will be a component that can be used for control of the system. In addition, Fourier analysis shows that the gain will not be linearly dependent on the duty cycle of the control system, and thus linearly variable pulse width modulation schemes are not viable to maintain control.

If this is truly the case, then, it was theorized that the controllers in the field that had experienced success with a subharmonic controller must have achieved it with the component of the control signal at the instability frequency. The gain of the controller at

the specific frequency was calculated with describing functions, and the predicted behavior of the system was described. Experiments were then performed with proportional gain pulse controllers. These compared the gain theorized to achieve control with the actual gain necessary to achieve control. Since calculation of the actual gain required with a linear phase shifter would have to take into account the physical parameters of the system and require an accurate model of the system, ratios between the actual gains were calculated and compared to ratios of the theoretical gains.

The results of these experiments were fairly conclusive. The expected gains matched very well with the theoretical gains. Comparing the ratios between different subharmonic signals and between linear phase shifters and non-subharmonic pulsed signals both produced near perfect results. However, there was some error seen when comparing the subharmonic signals to non-subharmonic signals or to the linear phase shifter results. This was attributed to nonlinearities in the system that caused a portion of the subharmonic component to be visible at the instability frequency.

After it was experimentally shown that the component of the control signal at the instability frequency was indeed responsible for control, fixed height pulse controllers were examined. These would result in a non-constant gain that depended on the amplitude of the limit cycle. Describing functions predicted that in these cases, the stable limit cycle level would be at the intersection of the describing function gain curve and the linear limit cycle amplitude curve. Experiments were done to confirm this, and it was shown that the limit cycle amplitude for a fixed height pulsed controller could be predicted. In addition, the use of variable subharmonic ratios and duty cycles to vary the gain of the controller was examined, and it was shown that these had the effects calculated theoretically as well.

A few miscellaneous experiments were run to investigate other aspects of these theories. Pacemakers were examined, since they have been used in other experiments. It was shown that pacemakers could influence the level of control. However the mechanism of this control was different, and in general the same level of control could not be achieved with pacemakers alone before the flame was blown off. Finally control with filtered subharmonic signals, which eliminated any component of control at the instability frequency, further validated the hypothesis that only the component at the

instability frequency was useful for control. It was shown that control could not be achieved using such a signal.

5.2 Future Work

The theories presented in this thesis provide a great number of paths to take for future work. A major improvement in pulsed control could be made by designing a pulsed fuel controller that would vary the pulse width and subharmonic ratio of the control signal dynamically to maintain the gain of the system at the optimal gain for control. This will also keep the gain of fixed height systems from tending towards infinity, which limits the effectiveness of control due to the ultimate gain of the system being reached. This would also have the benefit of reduced fuel consumption by only injecting fuel for the minimum time necessary. Actuator life could also be increased because the number of cycles of the fuel injector would be reduced.

The theoretical explanation for other mechanisms of control, such as pacemakers, could also be found. These explanations could enable further gains in fuel economy and emissions by reducing the initial instability level. This would therefore lower the control effort needed to eliminate the instability with the active controller.

Work should be continued on proportional fuel injectors. These would enable the gain of the system to be fixed at the optimum gain for control, and eliminate the problems fixed height injectors have with ultimate gain. It would allow the instability level to be reduced much further than what can be achieved even with an optimal fixed height controller, which will still suffer from an increased gain when the instability level is very low.

Finally, if proportional height fuel injectors can be designed, work should be done on pulse shaping. This would involve building a control pulse shape from the desired components of control. This could possibly allow a subharmonic component to be added for control, taking advantage of the nonlinearities of the system. Tests would have to be done to determine if this was more advantageous than using a regular linear controller or not, however.

There are many exciting possibilities in the world of pulsed fuel control. There will be further improvements in the area of stability and control, which will in turn lead to better performance, lower emissions, and improved fuel economy.

Bibliography

1 Annaswamy, A.M., Fleifil, M., Rumsey, J.W., Prasanth, R., Hathout, J.P., and Ghoneim, A.F., "Thermoacoustic Instability: Model-based Optimal Control Designs and Experimental Validation," *IEEE Transactions on Control Systems Technology*, Volume 8, Issue 6, Nov. 2000, pg. 905-918.

2 Culick, F. E. C., "A Note on Rayleigh's Criterion," *Combustion Science and Technology*, Vol. 56, 1987, p. 159-166.

3 Fannin, C. A., Saunders, W. R., and Vandsburger, U., *Influence Of Nonlinear Dynamics On Linear Controller Design For Instability Suppression In A Premixed Tube Combustor*, IEEE Conference on Control Applications – Proceedings, v. 2 1998, IEEE, Piscataway, NJ, USA, 98CB36104, p 1180-1184.

4 Heising, R., Lubarsky, E., Neumaier, M., Neumeier, Y., and Zinn, B. T., *Periodic Liquid Fuel Sprays Combustion Processes And Their Damping Of Combustion Instabilities*, AIAA 2000-1024, 38th Aerospace Sciences Meeting & Exhibit, Reno, NV, January 10-13, 2000.

5 Jones, C. M., Lee, J. G., and Santavicca, D. A., "Closed-Loop Active Control of Combustion Instabilities Using Subharmonic Secondary Fuel Injection," *Journal of Propulsion and Power*, Vol. 15, No. 4, July-August 1999.

6 Kim, K., Lee, J., Stenzler, J., and Santavicca, D., *Optimization of Active Control Systems for Suppressing Combustion Dynamics*, RTO Symposium on Active Control Technology for Enhanced Performance Operational Capabilities of Military Aircraft, Land Vehicles and Sea Vehicles, Braunschweig, Germany, May 8-11, 2000.

7 Lagimoniere, E., "The Design and Construction of a High Bandwidth Proportional Fuel Injection System for Liquid Fuel Active Combustion Control," [Thesis], Virginia Tech, etd-08222001-165330, 2001.

8 Mahmoud, H., Fleifil, M., Ghoneim, Z., and Ghoniem, A. F., "Active Control of Thermoacoustic Instability Using LQR Techniques," *Joint Power Generation Conference*, EC-Volume 5, 1997.

9 McManus, K. R., and Magill, J. C., "C*ontrol of thermo-acoustic instabilities using pulse width modulation*, IEEE Conference on Control Applications - Proceedings 1997, IEEE, Piscataway, NJ, USA, 97CH36055, p. 824-829.

10 McManus, K. R., Magill, J. C., and Miller, M. F., *Combustion Instability Suppression in Liquid-Fueled Combustors*, AIAA 98-0642, 36th Aerospace Sciences Meeting & Exhibit, Reno, NV, January 12-15, 1998.

11 Nord, Lars, "A Thermoacoustic Characterization of a Rijke-type Tube Combustor," [Thesis], Virginia Tech, etd-03102001-114618, 2000.

12 Richards, G. A., McMillan, M. M., Gemmen, R. S., Rogers, W. A., amd Cully, S. R., "Issues for Low-Emission, Fuel-Flexible Power Systems," *Progress in Energy and Combustion Science*, Vol. 27, No. 2, 2001, p. 141-169.

13 Richards, G. A., Yip, M. J., Robey, E., Cowell, L., and Rawlins, D., *Combustion oscillation control by cyclic fuel injection*, Proceedings of the International Gas Turbine and Aeroengine Congress and Exposition, Houston, TX, USA, June 5-8, 1995.

14 Saunders, W. R., Vaudrey, M. A., Eisenhower, B. A., Vandsburger, U., and Fannin, C. A., *Perspectives on Linear Compensator Designs for Active Combustion Control*, AIAA 99-0717, 37th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, January 11-14, 1999.

15 Vaudrey, M. A., and Saunders, W. R., *Control of Combustor Instabilities Using an Artificial Neural Network*, ASME Turbo Expo 2000, May 8-11, 2000, Munich, Germany.

16 Yu, K., Wilston, K. J., Parr, T. P., and Schadow, K. C., *An Experimental Study on Actively Controlled Dump Combustors*, NATO/RTO Active Control Symposium, Braunschweig, Germany, May 8-12, 2000.

Appendix A Controller Code

Some basic knowledge of C-programming is necessary to understand this section. All sections of the code will be explained in general, but some details may be left up to the reader. The code is shown in Section A.10, and will be referred to by the line numbers shown in the left hand column.

This code is written to perform control in real time. As such, it has only one sample period to perform each loop. The DSP board generates an interrupt every sample period (0.0001 seconds in this case), and this interrupt initiates a function call which reads the input from the A/D converter, performs the processing and decision making, and outputs a control signal through the D/A converter.

A.1 Variable declarations

The variables are declared in lines 18-77. Initially two header files are loaded. The first, brtenv.h, is included with the DS1103 board and contains function calls to read inputs from the A/D converters and output signals to the D/A converters. The second is math.h, the C mathematics library necessary to do standard math functions. After these declarations are the float variables. The Float64 data type is defined by the DSP board, and designates the variables to be 64-bit. Floats in general are signals which will be manipulated, or intermediate calculations using these signals. Variables of particular interest are mic1_in, which will be the value of the current sample, and control_output, which is the current value to be output. Other variables are described in the comments in the code. After the floats are defined, the integers (ints) are defined. These are in general counters, switches, and limits. For instance, the hold_count variable is a counter, which will be incremented when the processor is in the hold segment of a pulse. It will increment this counter until it reaches hold_samples, a limit, at which time it will switch to the next state of the processor by changing the value of state_switch, a switch variable.

A.2 Main() Function

The main() function is run by the processor at start up. The compiler requires it to be at the end of the code, thus its position from lines 365-389. This function begins by calculating the sample period from the sample frequency (defined as Samp_Freq=10000 Hz in this case). It then initializes the DSpace board with the init() function, included with the DSP board. It then specifies the channels to be read from the multiplexers on the board. A/D converter 1 will read channel 1, converter 2 will read channel 5, converter 3 channel 9, and converter 4 channel 13. It then sends a message to the DOS text screen saying the system has been started and begins the interrupt timer routine. The command RTLIB_SRT_START(period, isr_t0) causes the program to execute the interrupt service routine function, isr_t0, every sample period, defined as period above. After starting the interrupt service routine timer, the processor enters a while loop and simply checks its processor temperature.

A.3 Isr_t0 Initialization

The beginning of the interrupt service routine (lines 94-109) store the current values of the input, mic1_in, and piston output, piston_output, into variables for storage. Note at this point the value of the input has not yet been read from the A/D converters. After these previous values are stored, the current values are read. A/D converters 1 and 2 are started, and then the values for each are read. Note that as configured in main(), converter 1 will read channel 1 and converter 2 will read channel 5. The command ds1103_adc_read2 reads the value from 2 sequential A/D converters simultaneously, beginning with the number of the first argument (in this case the 1st A/D converter) and stores the values for the two inputs in the two pointers assigned as the second and third arguments. After this the code simply stores the main input into another variable for use if needed and increments the sample counter variable.

A.4 Overall Zero-Crossing Detector

The next section of the code, from lines 110-128, performs a series of calculations whenever a zero crossing is detected, regardless of the desired state of the controller. This allows the control panel to always display the estimated frequency of the input and the phase delay using the current delay_samples setting. A zero crossing is detected when the previous sample is less than zero and the current sample is greater than zero, or when the previous sample is greater than zero and the current sample is less than zero in conjunction with the 180° shift switch being enabled. This switch allows a quick addition of 180° of phase shift. When a zero crossing is seen, the sample counter variable is stored and then reset. Since the sample period is known, the frequency of the input can then be calculated as the reciprocal of the total time between zero crossings, which is the sample counter multiplied by the sample period. Similarly, the phase delay is calculated as the ratio of the number of delay samples to the number of samples in a period, multiplied by 360°.

A.5 Do Nothing State

The next overall section of the code is selected via a case statement. The variable STATE is set in the control panel to one of 4 possible control states. The first case, lines 131-141 when STATE=0, is trivial: the do nothing state. In this case the output of the controller, control_output, is set to zero, and various variables are reinitialized as needed.

A.6 Subharmonic Control State

The section of code from lines 142-320 contains the meat of the code in the processor and is the most difficult to understand. This section is switched to by the case statement when STATE=1, and performs pulsed control, subharmonic or not.

The first section of the code, lines 144-171, perform some pre-processing on the input signal. First the DC value of the input is subtracted from the signal. It was found that at extremely low signal levels a DC offset of a few millivolts was entering the converters and interfering with the zero crossing detection. Thus the DC offset of the

previous period of the signal was found and subtracted. Lines 148-157 are concerned with finding this offset, as they find the maximum positive and negative values of the raw signal. These values are then used in line 181 to find the average value of the previous sample when a zero crossing is detected and the period is complete. Lines 157-166 calculate the absolute value of the shifted input (from now on referred to as the input), and lines 167-171 use this absolute value to find the amplitude of the previous period of the input. This value is reset on zero crossing in line 271.

The next section of the subharmonic control section, lines 175-274, performs various operations and calculations when a zero crossing is detected. Many of these operations are not critical to control but are used for displaying information to the operator. The zero crossing is detected identically to above. When the zero crossing is seen, the effective gain is calculated for reference. The variables for the maximum negative and positive values of the input signal are reinitialized.

The following section of code, lines 186-232, is useful for display purposes. It was noticed that the estimated frequency was not very steady, and changed a lot due to the fact a period did not contain an exact integer number of samples. Thus one period might have 55 samples and the following period 56. This did cause some problems in control, as it was desired to perfectly reproduce a signal from one day to the next, and this was difficult with the estimated duty cycle and phase unsteady. To display constantly readable estimates, an average was taken. To do this, an array was declared which stored up to the previous 20 period lengths, as a number of samples per period. This array operated as a first-in-first-out array, with the oldest value being discarded when a zero crossing was detected. Thus the array pointer would move circularly through the array. The total number of samples of all periods in the array is then computed, and the total number is divided by the number of periods in the array to reach an average period length. This period length is then used to calculate the estimated frequency, estimated phase delay, and estimated duty cycle.

Finally at line 239 the control process is restarted. At line 239 it is decided whether or not this zero crossing is an appropriate one to send a pulse on or not. For instance, if third-subharmonic control is desired, every third zero crossing should cause a pulse to be sent. If the zero_cross count is greater than or equal to the desired

subharmonic ratio, freq_divider, the next state of the code is entered by setting state_switch=1. Also, counter variables for the pulse sample counts are reinitialized and the magnitude of the output is calculated. If the proportional magnitude control switch (prop_mag) is set to 1, the output is set to the gain value (mag_mult) multiplied by the amplitude of the previous period (max_mag). This causes a proportional pulse train signal to be output. Conversely, if the prop_mag switch is set to zero, the desired output is simply set to the mag_mult value, causing a fixed pulse height signal. If the system is using a fixed height signal, the if statements following the setting of the pulse height may be used to perform variable subharmonic ratio control. If this feature is turned on by setting freq_div_ch_sw to on, the effective gain of the system will be used to determine if a higher subharmonic ratio can be used or if a lower subharmonic ratio is necessary. The limits for these switches are user-definable via the control panel. Not much development was done on this section of the code.

If it has been decided that a pulse is necessary, the next section of the code, lines 275-287, perform the phase delay. When state_switch is set to 1, the delay counter is started. This is incremented every sample period. When the value of the counter, delay_count, reaches the desired number of delay samples, delay_samples, the system will go on to the next stage by setting state_switch=2. At this time, the system also begins the pulse output by setting control_output equal to the positive value of the desired pulse height.

The system is now outputting a pulse, and lines 288-300 decide when to turn it off. This is very similar to the previous section. A hold counter is started upon entry into this section, and when the hold_count variable reaches the desired number of hold samples, hold_samples, turns off the pulse by setting the output to the negative desired height of the pulse and returning state_switch=0. This will cause the system to return to the state of looking for zero crossings. It should be noted that too long of a combined delay and hold time can cause errors if they are longer than the sample period, because the system will not be in a state to switch on zero crossings if it is still in the hold or delay period.

Finally, lines 301-315 were added to allow for experiments using pulsed fuel control. A piston was used to force fuel into the combustor. As it needed a position input

to realize a desired velocity output, it was necessary to send the piston the integral of the desired velocity. Lines 310-315 perform this integration. First the DC offset of the output is calculated by using the estimated duty cycle. This offset is subtracted from the output, which is then integrated to obtain the piston_output. A check valve is used to prevent the piston from back-driving the supply system, and lines 300-309 close the check valve when a pulse is being output and open it otherwise. This is simply done by setting the check_valve_output to 1 for positive output signals and 0 for negative output signals.

A.7 Linear Phase Shifter

The next section of code, lines 320-335 for the case STATE=2, implement a standard linear phase shifter. This was accomplished by circling through an array. The first delay_samples number of values in an array was used, depending on the desired phase delay. An array pointer simply circled through the array. The current value of the input was read into the array at the position of the pointer, and the value of the array at the position immediately following the pointer was read as the output. This causes the oldest value of the array to always be the output value. Therefore the phase delay is simply the age of the array, or the number of cells that the pointer circles through. This causes the input signal to be delayed by a set number of samples and then output.

A.8 Echo

For the final value of the case statement, STATE=3 in lines 336-348, the input of the controller is simply sent to the output. This can be useful in debugging to ensure that the same signal is entering the controller as is shown on other devices such as oscilloscopes.

A.9 Output Segment

All the processing on the input has now been done and the desired output values have been decided. It is finally time to perform the output. Lines 349-360 perform this

output. Lines 351-355 allow any second input to be added to the output. This was extremely useful in examining probe signals and open-loop control. The ds1103_dac_write2 command was then used to output the control signal, a DC-shifted valued of the control signal with no DC component, a check valve output, and a piston output. Thus ends the explanation of the C code used to implement the controllers used in this thesis.

A.10 Code

```
1  /*********************************************************
2
3  Subharmonic square wave
4
5  Generates subharmonic square wave at +- out_mag at frequency
6  divisions of the input wave
7
8  Capability to delay from zero crossing
9  Added variable duty cycle
10 Frequency Estimation
11 Phase Delay
12
13 Matthew Carson
14 11/14/2000
15
16 *********************************************************************/
17
18 #include <brtenv.h>
19 #include <math.h>
20
21 Float64 Samp_Freq = 10000.0;                /* sampling frequency */
22 Float64 period;                             /* sampling period */
23 Float64 mic1_in;                            /* filtered, corrected input */
24 Float64 abs_mic1_in;                        /* absolute value of mic1_in */
25 Float64 input2;                            /* second input to be added to output */
26 Float64 input1;                            /* unfiltered input */
27 Float64 last_mic1_in;                      /* stored last mic input */
28 Float64 max_neg = 0.0;                     /* maximum negative value of input */
29 Float64 max_pos = 0.0;                     /* maximum positive value of input */
30 Float64 mic1_avg;                          /* average value of last period */
31 Float64 input_sig;                         /* input signal used */
32 Float64 control_output = 0.0;              /* output variable */
33 Float64 dc_offset = 0.0;                   /* DC offset of output signal */
34 Float64 shifted_output = 0.0;              /* DC offset subtracted from output */
35 Float64 last_shifted_output = 0.0;         /* previous value of shifted output */
36 Float64 check_valve_output = 0.0;          /* check valve output */
37 Float64 piston_output = 0.0;               /* output for piston */
38 Float64 last_piston_output = 0.0;          /* previous value of piston output */
39 Float64 integral_constant = 200;           /* multiplier for integral */
40 Float64 out_mag = 0.5;                     /* magnitude of output (mas 1) */
41 Float64 zero_cross = 0.0;                  /* zero crossing counter */
42 Float64 est_freq;                          /* estimated input frequency */
43 Float64 est_phase_delay;                   /* estimated output phase delay */
44 Float64 est_duty_cycle;                    /* estimated output duty cycle */
45 Float64 samp_store[200];                   /* store samples for phase delay */
46 Float64 max_mag = 0.0;                     /* maximum magnitude of input */
47 Float64 mag_mult = 1.0;                    /* proportional constant for magnitude */
48 Float64 eff_gain;                          /* effective gain of non-proportional signal */
49 Float64 freq_div_sw_lo = 2.7;              /* switch level for changing to lower freq div */
50 Float64 freq_div_sw_hi = 4.5;              /* switch level for changing to lower freq div */
51
52 int STATE = 1;                             /* state of processor */
53 int delay_samples = 17;                    /* samples to delay output from zero-crossing */
54 int hold_samples = 28;                     /* samples to hold output high */
55 int delay90_samples = 0;                   /* samples to delay for 90 degree phase shift */
56 int hold90_samples = 0;                    /* samples to hold for 90 degree phase shift 50%
d.c.*/
```

```
57 int freq_divider = 1;                        /* frequency division factor */
58 int delay_count = 0;                         /* count variable for delay */
59 int hold_count = 0;                          /* count variable for hold */
60 int state_switch = 0;                        /* state variable for different phases */
61 int samp_count = 0;                          /* number of samples in period */
62 int samp_count_array[21];                    /* array of period lengths */
63 int samp_count_index = 1;                    /* count variable for period array */
64 int min_good_samp_count = 50;                /* minimum counter value for a valid period */
65 int cross_good = 0;                          /* denotes good zero crossing */
66 int average_index = 0;                       /* index for averaging frequency */
67 int average_count = 0;                       /* count variable for averaging freq */
68 int input_count = 0;                         /* count for input for phase delay */
69 int output_count = 0;                        /* count for output for phase delay */
70 int array_full = 0;                          /* switch for period length array full */
71 int last_samp_count = 0;                     /* storage for last period length */
72 int swap_180 = 1;                            /* switch for 180 degree phase shift */
73 int total_time = 0;                          /* total time of periods in array */
74 int prop_mag = 1;                            /* whether to use proportional magnitude */
75 int bp_on = 0;                               /* turn on band pass filter */
76 int add_input2 = 0;                          /* switch to add second input to output */
77 int freq_div_ch_sw = 0;                      /* switch to turn on prop freq division */
78
79 /*--------------------------------------------------------------------*/
80
81
82 /* variables for execution time profiling */
83 #define TMR0 0                               /* timer0 definition */
84 Float32 exec_time;                             /* execution time */
85 unsigned long count0;                        /* timer0 time count */
86
87
88
89
90 /*--------------------------------------------------------------------*/
91
92 void isr_t0()                                /* timer0 interrupt service routine */
93 {
94   RTLIB_SRT_ISR_BEGIN();                                  /* overload check */
95   host_service(1, 0);                                      /* TRACE service */
96
97   RTLIB_TIC_START();                              /* start time measurement */
98
99   last_mic1_in = mic1_in;                     /* store last input */
100  last_piston_output = piston_output;         /* store last piston output */
101  ds1103_adc_start(DS1103_ADC1 | DS1103_ADC2);
102
103  ds1103_adc_read2(1,&input_sig,&input2);
104
105  mic1_in = input_sig;
106  samp_count++;                       /* increment period length counter */
107  /* the following code will run if there is a zero crossing with a positive slope
108     or if there is a zero crossing with a negative slope and the swap_180 flag is set */
109
110  if (((last_mic1_in <= 0) && (mic1_in > 0) && (swap_180 == 0)) || ((last_mic1_in >= 0) && (mic1_in
< 0) && (swap_180 == 1)) && (samp_count > min_good_samp_count))
111  {
112     last_samp_count = samp_count;    /* store last period length */
113     total_time = 0;                  /* reset total time storage variable */
114     delay_count = 0;                 /* reset delay counter */
115     samp_count = 0;                  /* reset period length counter */
116     cross_good = 1;
117     average_index = 1;               /* reset averaging index */
118
119     est_freq=1/(last_samp_count*period);
120
121      /* the following estimates the phase delay of the output as the number of
122        delay samples divided by the period length *360 degrees */
123     if (STATE != 1)
124     {
125        est_phase_delay = (delay_samples+1)*360/last_samp_count;
126     }
127   }
128
129   switch (STATE)                             /* choose what to do depending on STATE */
130   {
131      case 0:                                 /* case 0 do nothing */
132      {
133         /* reinitialize all variables */
134         control_output = 0;
135         zero_cross = 0;
136         state_switch = 0;
```

```
137         average_index = 1;
138
139         break;
140      }
141
142      case 1:     /* subharmonic control, with check valve and piston outputs */
143      {
144         mic1_in = input_sig - mic1_avg;
145
146         /* find maximum magnitude of input */
147
148         if (input_sig < max_neg)
149         {
150            max_neg = input_sig;
151         }
152
153         if (input_sig > max_pos)
154         {
155            max_pos = input_sig;
156         }
157
158         if (mic1_in < 0)
159         {
160            abs_mic1_in = (-1)*(mic1_in);
161         }
162         else
163         {
164            abs_mic1_in = mic1_in;
165         }
166
167         if (abs_mic1_in > max_mag)
168         {
169            max_mag=abs_mic1_in;
170         }
171
172         /* the following code will run if there is a zero crossing with a positive slope
173            or if there is a zero crossing with a negative slope and the swap_180 flag is set */
174
175         if (((last_mic1_in <= 0) && (mic1_in > 0) && (swap_180 == 0)) || ((last_mic1_in >= 0) &&
(mic1_in < 0) && (swap_180 == 1)) && (cross_good == 1))
176         {
177            cross_good = 0;
178            zero_cross++;                  /* zero crossing - increment counter */
179
180            eff_gain = (2*mag_mult)/(freq_divider*max_mag);
181            mic1_avg = (max_pos + max_neg) / 2.0;
182
183            max_neg = 10;                  /* reset maximum negative value */
184            max_pos = -10;                 /* reset maximum positive value */
185
186            /* the following inserts the last period length into the array */
187            samp_count_array[samp_count_index] = last_samp_count;
188
189            /* the following decides how many samples are currently in the array and
190               thus how many to average.  For the first 20 periods this will count up
191               then will stay at 20. */
192            if (array_full == 1)
193            {
194               average_count = 20;
195            }
196            else
197            {
198               average_count = samp_count_index;
199            }
200
201            /* the following loop calculates the total number of samples in the last
202               average_count periods by adding the periods from those elements of the
203               array */
204            while (average_index <= average_count)
205            {
206               total_time += samp_count_array[average_index];
207               average_index++;
208            }
209
210            /* the following calculates the frequency of the input as the total time
211               in the last average_count periods divided by the number of periods */
212            est_freq=1/(total_time*period/average_count);
213
214
215            /* the following estimates the duty cycle of the output as the number of
216               hold samples divided by the period length *100% */
```

```
217            est_duty_cycle = (hold_samples*100)/((total_time/average_count)*freq_divider);
218
219            /* the following estimates the phase delay of the output as the number of
220              delay samples divided by the period length *360 degrees */
221           est_phase_delay = (delay_samples+1)*360/(total_time/average_count);
222
223            /* the following will reset the index for the period length array if it is higher
224               than the number of elements in the array, and then increment it.  This will
225               result in the oldest period length always being thrown out as the index
226               cycles through the array */
227            if (samp_count_index >= 20)
228            {
229               array_full = 1;
230               samp_count_index = 0;
231            }
232            samp_count_index++;
233
234            /* finally if the current zero crossing is a zero crossing that we want to
235               send a pulse on, go the next phase (state_switch = 1 [delay]).  This is decided
236               if the current number of zero crossings is greater than the frequency divider
237               variable.  i.e. for a frequency division of three we will send a pulse on every
238               third zero crossing */
239            if (zero_cross >= freq_divider)
240            {
241               state_switch = 1;             /* go to next phase */
242               delay_count = 0;              /* reset delay counter */
243               zero_cross = 0;               /* reset zero crossing counter */
244               hold_count = 0;               /* reset hold counter */
245               last_piston_output = 0;       /* reset integral - piston output */
246
247               /* if proportional magnitude control is turned on set output magnitude
248                  to constant times maximum input magnitude */
249               if (prop_mag==1)
250               {
251                  out_mag=max_mag*mag_mult;
252               }
253               else
254               {
255                  out_mag=mag_mult;
256                  if (freq_div_ch_sw == 1)
257                  {
258                     if (eff_gain > freq_div_sw_hi)
259                     {
260                        freq_divider++;
261                     }
262                     if (eff_gain < freq_div_sw_lo)
263                     {
264                        if (freq_divider != 1)
265                        {
266                           freq_divider = freq_divider - 1;
267                        }
268                     }
269                  }
270               }
271               max_mag = 0;                  /* reset maximum magnitude */
272             }
273
274         }
275         if (state_switch == 1)              /* delay phase */
276         {
277
278            /* the following will do nothing until the proper delay has been achieved.
279               It will then set the output to the positive value of out_mag and go to
280               the next phase (state_switch = 2 [hold]) */
281            if (delay_count >= delay_samples)
282            {
283               control_output = out_mag;     /* set output to positive */
284               state_switch = 2;             /* go to next phase */
285            }
286            delay_count++;                   /* increment delay counter */
287         }
288         if (state_switch == 2)              /* hold phase */
289         {
290            /* the following will do nothing until the proper hold time has been achieved.
291               It will then set the output to the negative value of out_mag and go back to
292               doing nothing which will wait for the next zero crossing */
293            if (hold_count >= hold_samples)
294            {
295               control_output = (-1)*out_mag;/* set output to negative */
296               state_switch = 0;             /* go back to first stage */
297            }
```

```
298            hold_count++;                    /* increment hold counter */
299         }
300
301         if (control_output > 0)
302         {
303            check_valve_output = 1;
304         }
305         else
306         {
307            check_valve_output = 0;
308         }
309
310         dc_offset = out_mag * ((2 * est_duty_cycle/100) - 1);
311
312         last_shifted_output = shifted_output;
313         shifted_output = control_output - dc_offset;
314
315         piston_output = last_piston_output + integral_constant*period/2.0*(shifted_output +
last_shifted_output);
316
317      break;
318      }
319
320      case 2:     /* pure phase delay */
321      {
322         samp_store[input_count] = mic1_in;
323         output_count = input_count+1;
324         if (output_count > delay_samples) output_count = 0;
325         control_output = mag_mult * samp_store[output_count];
326         if (array_full != 1) control_output = 0;
327         input_count++;
328         if (input_count > delay_samples)
329         {
330            input_count = 0;
331            array_full = 1;
332         }
333         break;
334      }
335
336      case 3:                              /* case 3 echo input*/
337      {
338         control_output = mic1_in;           /* set output to input */
339
340         /* reinitialize all variables */
341         zero_cross = 0;
342         state_switch = 0;
343         average_index = 1;
344
345         /* output current control_output */
346         break;
347      }
348   }
349  /*Output_Chan(control_output);*/
350
351  if (add_input2 == 1)
352  {
353     control_output = control_output + input2;
354  }
355
356  ds1103_dac_write2(1, control_output, shifted_output);
357  ds1103_dac_write2(3, piston_output, check_valve_output);
358
359  exec_time = RTLIB_TIC_READ();
360  RTLIB_SRT_ISR_END();                                  /* overload check */
361}
362
363/*---------------------------------------------------------------------*/
364
365main()
366{
367  Int16 temperature;
368
369  period = 1.0/Samp_Freq;                   /* calculate period */
370
371  init();                              /* DS1103 and RTLib1103 initialization */
372
373  ds1103_adc_mux_all(1,5,9,13);       /* specify channels to be read from multiplexer */
374
375  msg_info_set(MSG_SM_RTLIB, 0, "System started.");
376
377  RTLIB_SRT_START(period, isr_t0);          /* start sample rate timer */
```

```
378
379  while(1)
380  {
381
382    RTLIB_BACKGROUND_SERVICE();                       /* background service */
383
384    /* writes the CPU-temperature into the Config Section */
385    ds1103_cpu_temp_read(&temperature);
386
387  }
388
```

Appendix B ControlDesk Interface

Creating the user interface for the subharmonic controller was fairly straightforward. First, any variables to be monitored or controlled in the interface were entered into a trace file, which was compiled with the C code. A user interface was then created in the ControlDesk 2.0 software that allowed the user to change and monitor the desired variables. This will then provide a graphical user interface so that a complete understanding of the inner workings of the C code is not necessary to use the controller.

B.1 Trace File

The trace file is a file defined specifically for use with DSpace's ControlDesk software. It tells the interface which variables in the processor to monitor. These variables can be entered in the trace file as groups, so that a tree will be generated in the interface for finding the variables to use. For instance, all inputs are grouped together. In addition, the variables can each be given an alias, as not all users will be familiar with the variables used in the C code. These aliases can give better descriptions of the variables. Finally, the data type for each variable is given. This allows the interface to properly display the data in the variables. For instance, a variable may be declared as a 64-bit float. Declaring the variable as a 32-bit float in the trace file will cause erroneous data to be displayed on the control panel. The trace file for the subharmonic controller, subharm.trc, is shown below. This file is combined at compile time with the subharm.c file to generate an object file downloadable to the DSpace board, subharm.ppc.

```
sampling_period = 1.0E-4

_author                 "Matthew Carson"
_gendate                "04/16/2001"
_description            "Generate subharmonic control signals"


-- signals available for TRACE
--
--  signal name         type  address
```

```
group "Inputs"
   input_sig
   {
      type:   flt(64,IEEE)
      alias:  "Actual Input"
   }

   mic1_in
   {
      type:   flt(64,IEEE)
      alias:  "Current Input"
   }

   last_mic1_in
   {
      type:   flt(64,IEEE)
      alias:  "Previous Input"
   }

   abs_mic1_in
   {
      type:   flt(64,IEEE)
      alias:  "Absolute Value of Input"
   }

   input1
   {
      type:   flt(64,IEEE)
      alias:  "Filtered Input"
   }

   input2
   {
      type:   flt(64,IEEE)
      alias:  "Input 2"
   }

   max_mag
   {
      type:   flt(64,IEEE)
      alias:  "Maximum Value of Previous Period"
   }
endgroup

group "Outputs"
   control_output
   {
      type:   flt(64,IEEE)
      alias:  "Output"
   }

   check_valve_output
   {
      type:   flt(64,IEEE)
      alias:  "Check Valve Output"
   }


   out_mag
   {
      type:   flt(64,IEEE)
      alias:  "Magnitude of Output"
   }

   shifted_output
   {
      type:   flt(64,IEEE)
      alias:  "Magnitude of Output"
   }

   check_valve_output
```

```
   {
      type:   flt(64,IEEE)
      alias:  "Check Valve Output"
   }

   piston_output
   {
      type:   flt(64,IEEE)
      alias:  "Piston Output"
   }

endgroup

group "Operating Parameters"
   group "Parameters"
      freq_divider
      {
         type:   int
         alias:  "Frequency Division"
      }

      delay_samples
      {
         type:   int
         alias:  "Number of Delay Samples"
      }

      hold_samples
      {
         type:   int
         alias:  "Number of Hold Samples"
      }

      mag_mult
      {
         type:   flt(64,IEEE)
         alias:  "Proportional Magnitude Multiplier // Constant Output"
      }

      freq_div_sw_hi
      {
         type:   flt(64,IEEE)
         alias:  "Frequency Division Hi Switch Limit"
      }

      freq_div_sw_lo
      {
         type:   flt(64,IEEE)
         alias:  "Frequency Division Lo Switch Limit"
      }

      integral_constant
      {
         type:   flt(64,IEEE)
         alias:  "Constant Multiplier for Integration"
      }
   endgroup

   group "Switches"
      swap_180
      {
         type:   int
         alias:  "Rising Edge // Falling Edge Count"
      }

      prop_mag
      {
         type:   int
         alias:  "Use Proportional Magnitude Switch"
      }
```

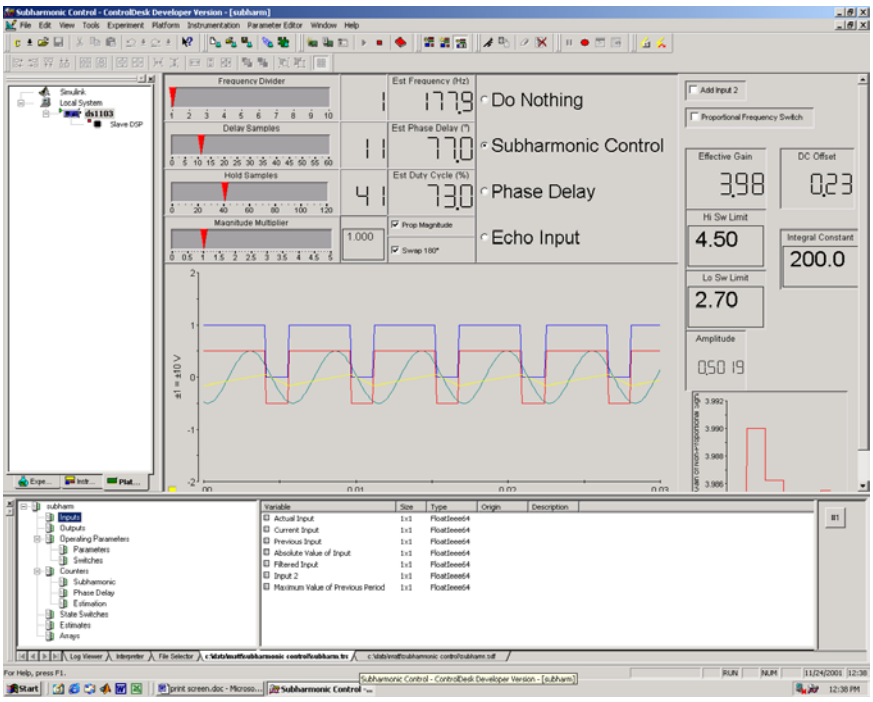
```
      bp_on
      {
         type:   int
         alias:  "Bandpass Filter Switch"
      }

      add_input2
      {
         type:   int
         alias:  "Add Second Input Switch"
      }

      freq_div_ch_sw
      {
         type:   int
         alias:  "Frequency Divider Change Switch"
      }
   endgroup
endgroup

group "Counters"
   group "Subharmonic"
      zero_cross
      {
         type:   flt(64,IEEE)
         alias:  "Zero Crossing Counter"
      }

      delay_count
      {
         type:   int
         alias:  "Delay Samples Counter"
      }

      hold_count
      {
         type:   int
         alias:  "Hold Samples Counter"
      }
   endgroup

   group "Phase Delay"
      input_count
      {
         type:   int
         alias:  "Input Array Index for Simple Phase Shift"
      }

      output_count
      {
         type:   int
         alias:  "Output Array Index for Simple Phase Shift"
      }
   endgroup

   group "Estimation"
      max_neg
      {
         type:   flt(64,IEEE)
         alias:  "Maximum negative value of input"
      }

      max_pos
      {
         type:   flt(64,IEEE)
         alias:  "Maximum positive value of input"
      }

      mic1_avg
      {
         type:   flt(64,IEEE)
```

```
         alias:  "Average value of input"
      }
      samp_count
      {
         type:   int
         alias:  "Number of Samples in Current Period"
      }

      last_samp_count
      {
         type:   int
         alias:  "Number of Samples in Previous Period"
      }

      samp_count_index
      {
         type:   int
         alias:  "Count Variable for Period Length Array"
      }

      total_time
      {
         type:   int
         alias:  "Total Time of Periods in Array"
      }

      average_index
      {
         type:   int
         alias:  "Count Variable for Frequency Averaging Index"
      }

      average_count
      {
         type:   int
         alias:  "Number of Periods to Average"
      }

      array_full
      {
         type:   int
         alias:  "Array Full Flag"
      }

   endgroup
endgroup

group "State Switches"
   STATE
   {
      type:   int
      alias:  "Control State"
   }

   state_switch
   {
      type:   int
      alias:  "Subharmonic State"
   }
endgroup

group "Estimates"
   est_freq
   {
      type:   flt(64,IEEE)
      alias:  "Estimated Frequency"
   }

   est_phase_delay
   {
      type:   flt(64,IEEE)
```

```
      alias:  "Estimated Phase Delay"
   }

   est_duty_cycle
   {
      type:   flt(64,IEEE)
      alias:  "Estimated Duty Cycle"
   }

   eff_gain
   {
      type:   flt(64,IEEE)
      alias:  "Effective Gain of Non-Proportional Signal"
   }

   dc_offset
   {
      type:   flt(64,IEEE)
      alias:  "DC Offset of signal"
   }
endgroup

group "Arrays"
   samp_count_array[1]     int
   samp_count_array[2]     int
   samp_count_array[3]     int
   samp_count_array[4]     int
   samp_count_array[5]     int
   samp_count_array[6]     int
   samp_count_array[7]     int
   samp_count_array[8]     int
   samp_count_array[9]     int
   samp_count_array[10]    int
   samp_count_array[11]    int
   samp_count_array[12]    int
   samp_count_array[13]    int
   samp_count_array[14]    int
   samp_count_array[15]    int
   samp_count_array[16]    int
   samp_count_array[17]    int
   samp_count_array[18]    int
   samp_count_array[19]    int
   samp_count_array[20]    int
   samp_store[0]           flt(64,IEEE)
   samp_store[1]           flt(64,IEEE)
   samp_store[2]           flt(64,IEEE)
   samp_store[3]           flt(64,IEEE)
   samp_store[4]           flt(64,IEEE)
   samp_store[5]           flt(64,IEEE)
   samp_store[6]           flt(64,IEEE)
endgroup
```

B.2 ControlDesk Interface

The variables shown above are then used to create the user interface. The variables will be shown in the trace file window of the ControlDesk interface as shown below.

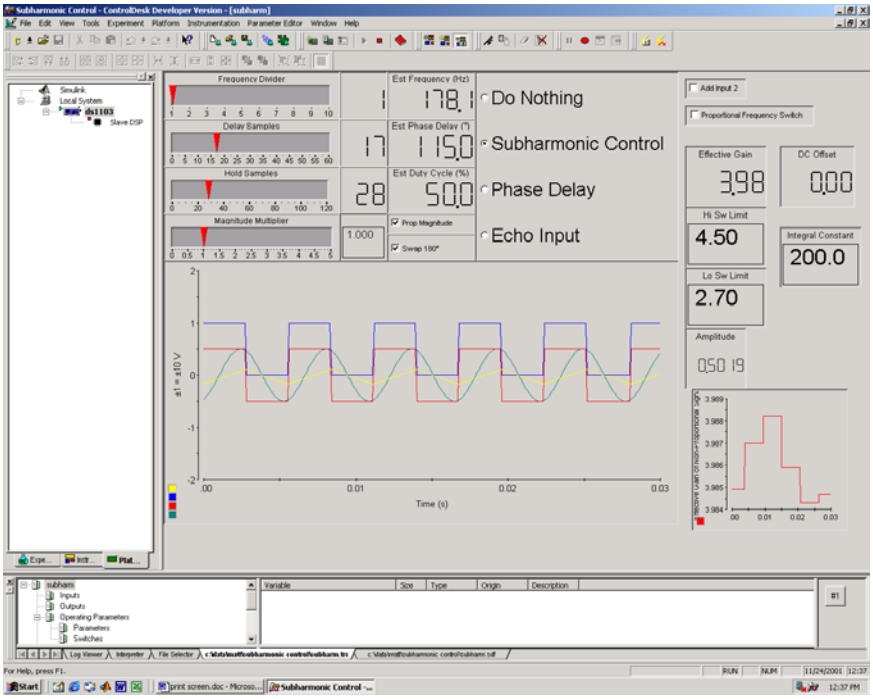


It can be seen that the tree of groups is on the bottom left window of the screen. Since "Inputs" is currently selected, all variables, shown by their aliases, are shown in the window on the bottom right of the screen. To use any of these variables in the ControlDesk interface, simply drag and drop it onto the desired control item. For instance, to create the plot of the input, first click on the "plotter" button and drag open a window of the desired size. Then drag the "Current Input" variable to the plotter and drop it. Similar actions can be taken for other display windows or for any controls such as entry boxes or slider bars.

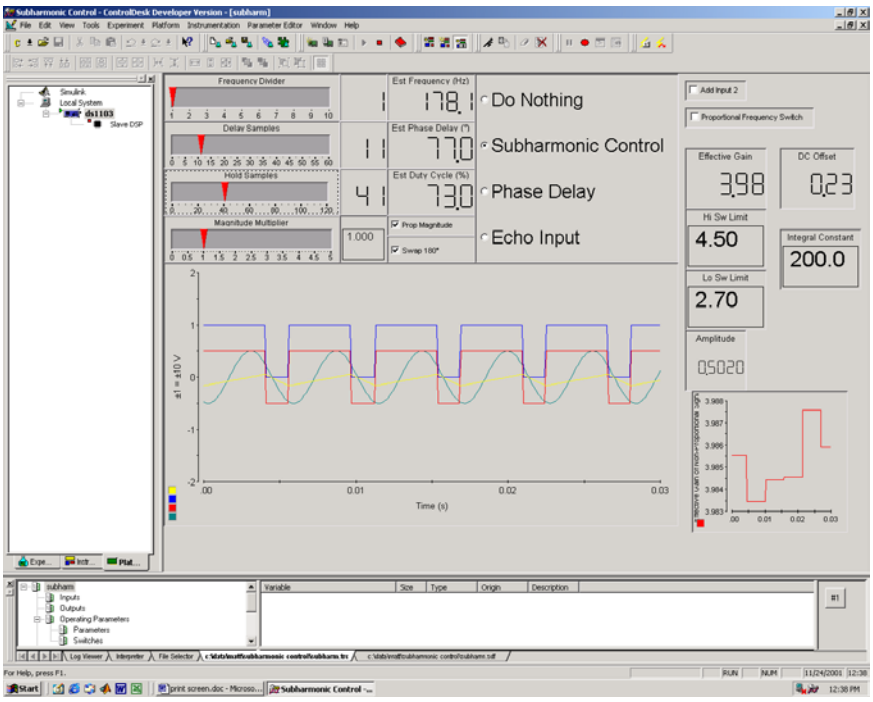
The interface is then built up until all desired items are complete. The program is automatically downloaded to the processor when it is compiled. If the program has not been compiled in the current power cycle, it must be downloaded to the processor by dragging the object file to the DS1103 board in the upper left "Platform" tab. The interface is then started by entering animation mode, a button on the toolbar at the top of the screen.

The interface is now ready to be used. Some examples of its use are shown below. Upon startup, the default values for all control variables are entered, which are the initial variable declaration variables defined in the C code. The default control state is a 1/1 subharmonic control signal with a 50% duty cycle (assuming 178 Hz instability)

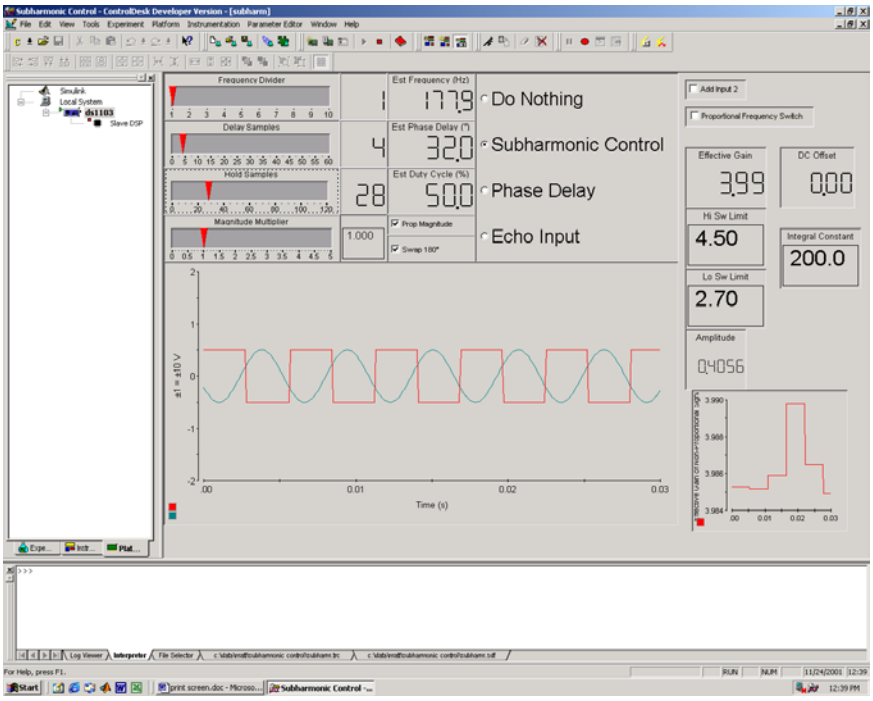
and a 115° phase shift to the start of the pulse from a negative going zero crossing (note that the "Swap 180°" box is checked). This controller uses proportional control (denoted by the checked "Prop Magnitude" switch) with a gain of 1.00. This state is shown below, and shows the input, the standard output, and the check valve and piston (integrated) outputs.



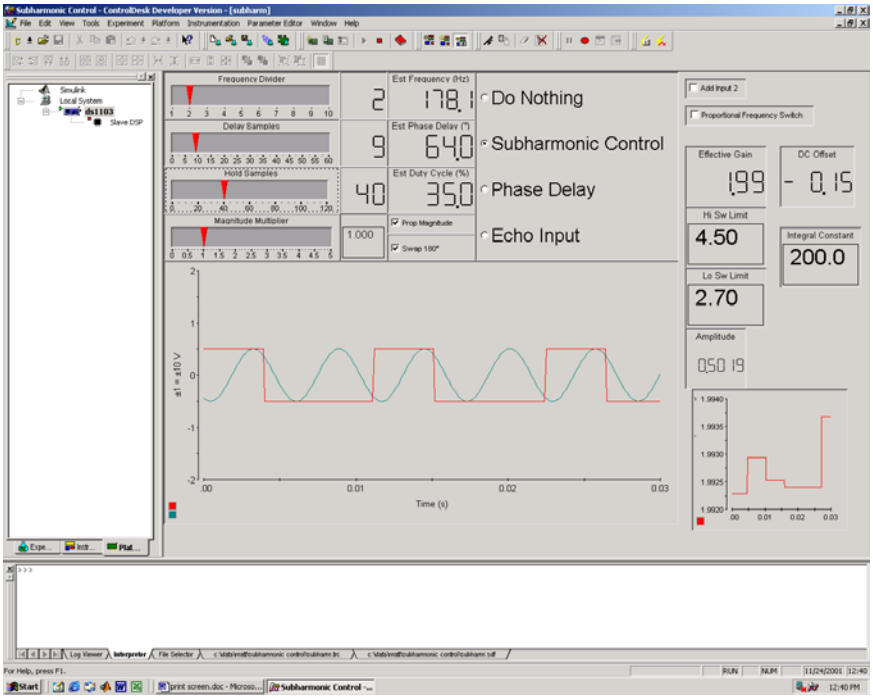
Moving the "Hold Samples" slider bar, as shown below, can change the duty cycle. In this case, it increases the duty cycle to 73%. Similarly, moving the "Delay Samples" slider bar changes the phase delay. In this case the phase shift is changed to 77°.



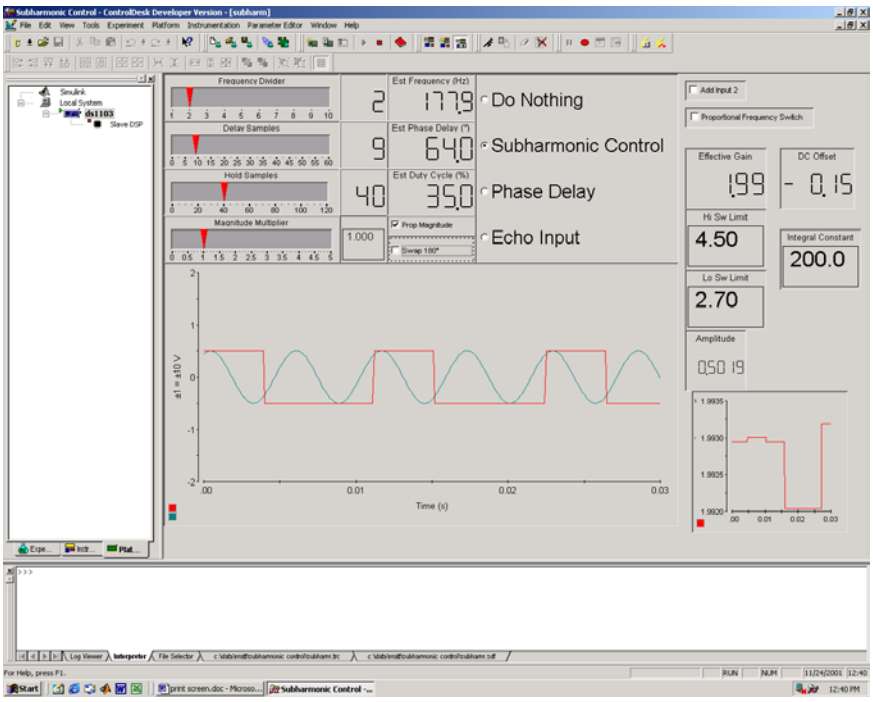
Next, for acoustic control, the check valve signal and the piston output signal can be deleted for clarity. This is done by clicking on the box corresponding to the color of these signals and pressing "Delete." This cleans up the screen as shown below.



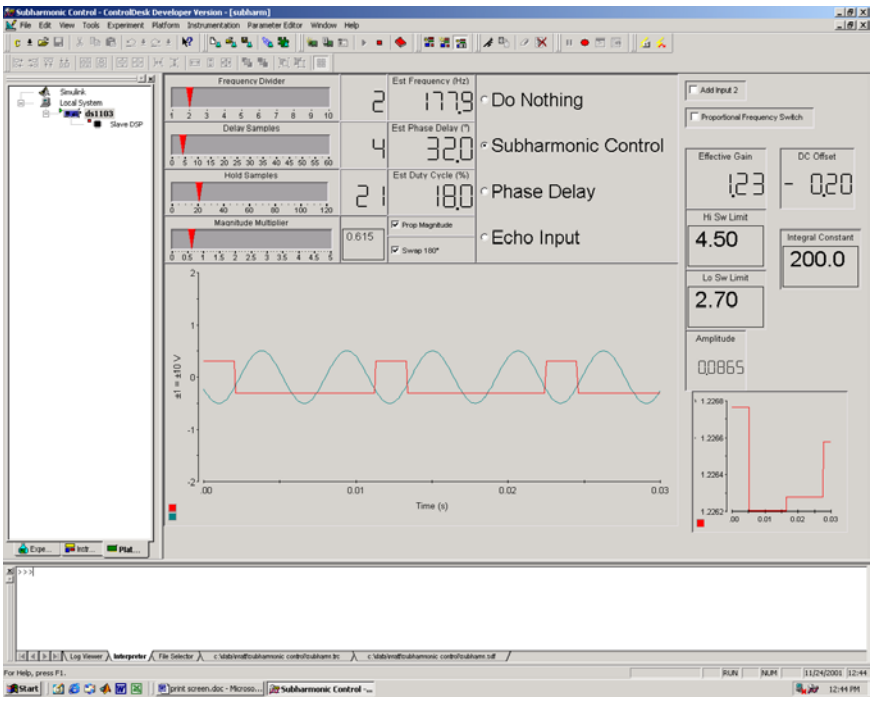
Subharmonic control can be initiated by dragging the “Frequency Divider” slider bar. As seen below, a setting of 2 causes pulses to be sent only every other period.



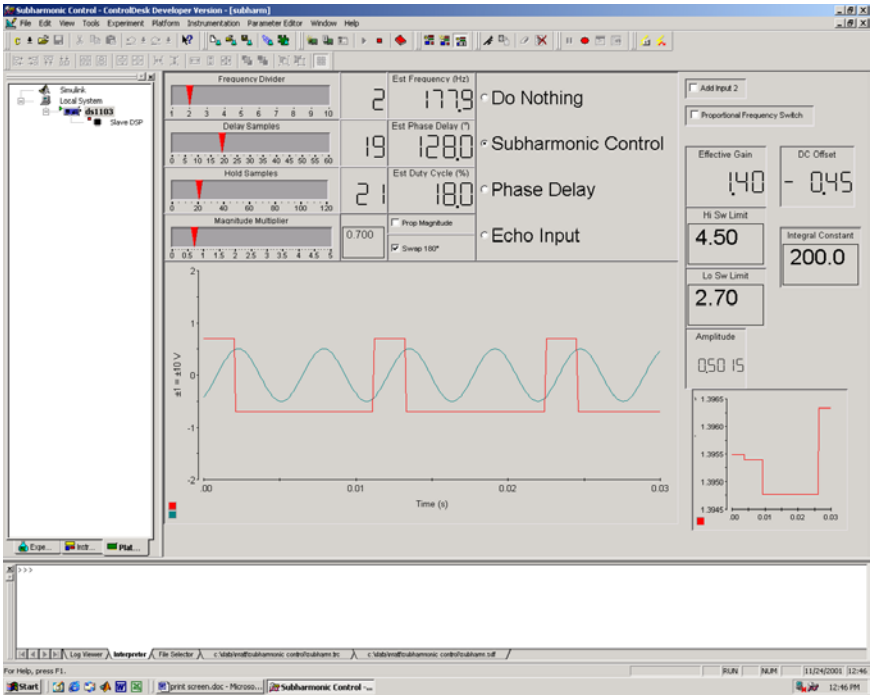
Unselecting the “Swap 180°” check box will simply cause the pulses to be sent on a delay after the positive going zero crossing. The controller below is identical to the one above except that the 180° shift option has been turned off.



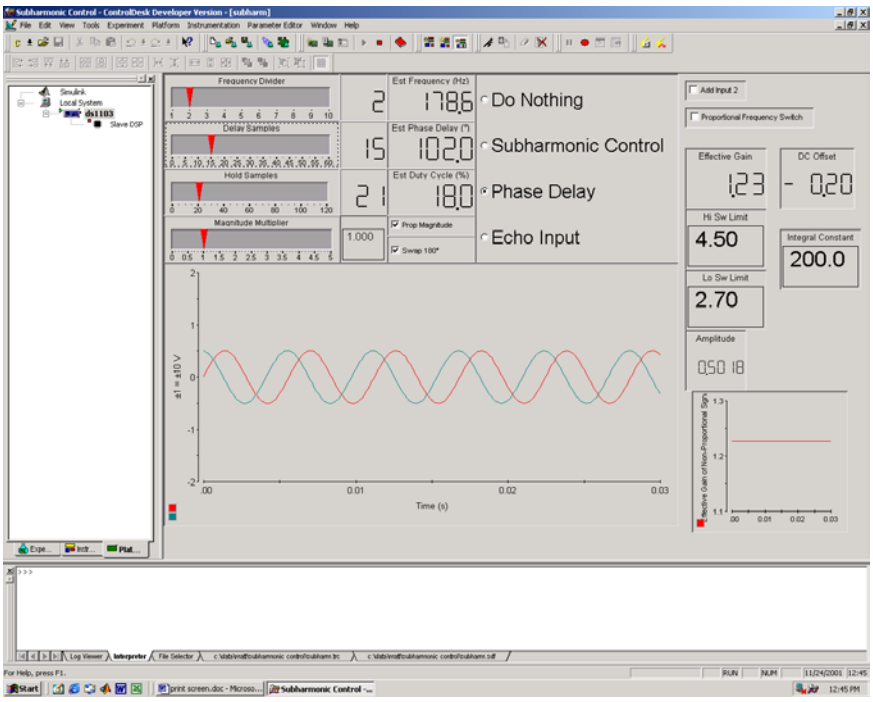
Next moving the "Magnitude Multiplier" slider bar or entering a gain value in the box accompanying the slider bar can change the height of the pulses proportionally to the height of the input. This is shown below, where the gain was reduced to 0.615.



A fixed pulse height can be entered in the same way by unselecting the “Prop Magnitude” check box.

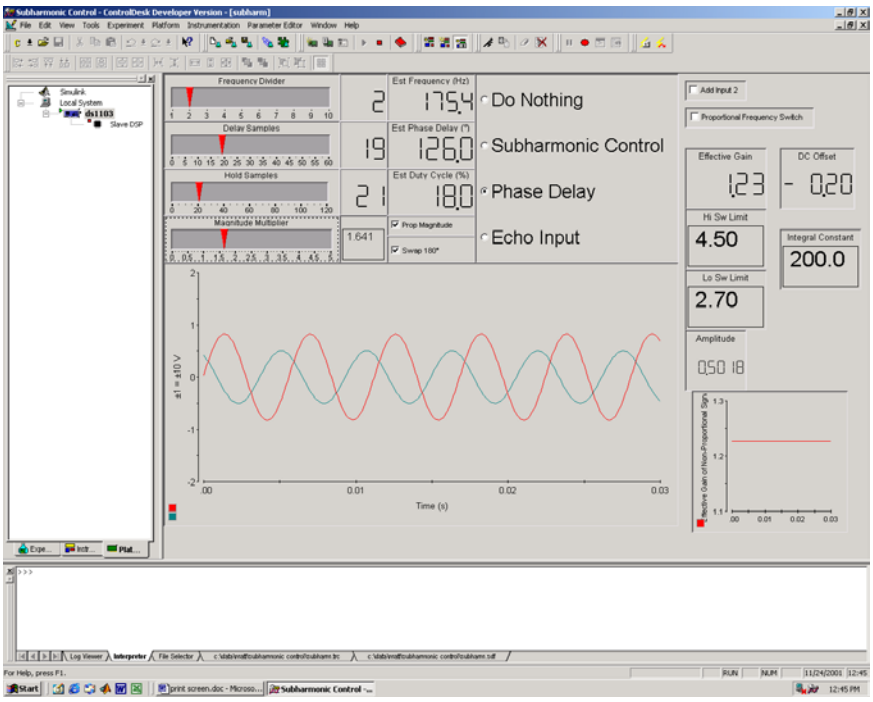


Finally, a linear phase shifter is also available for use by selecting the “Phase Delay” option in the upper center of the screen. This causes a pure delay to be used, as shown below.



The gain of this phase shifter can be changed as well by moving the "Magnitude Multiplier" slider or entering a gain in the box next to it as seen in the next figure.

Although not discussed in depth, the boxes on the right of the screen give more useful information. The plotter display is used to show a second input if one exists. It is added if the "Add Input 2" check box is selected. (It was not used for these figure generations.) The amplitude of the input signal is shown, as well as the DC offset. The effective gain is shown for subharmonic signals and is calculated as the output over the input, taking into account the duty cycle and subharmonic ratio. The two boxes below the effective gain are used as limits for the effective gain for the subharmonic ratio-switching algorithm, which is turned on by the "Proportional Frequency Switch" check box.


Subharmonic Control - ControlDesk Developer Version - [subharm]
Frequency Divider
Delay Samples
Hold Samples
Magnitude Multiplier
Est Frequency (Hz)
Est Phase Delay (°)
Est Duty Cycle (%)
Do Nothing
Subharmonic Control
Phase Delay
Echo Input
Prop Magnitude
Swap 180°
Add Input 2
Proportional Frequency Switch
Effective Gain
DC Offset
Hi Sw Limit
Integral Constant
Lo Sw Limit
Amplitude
Time (s)

Vita

James Matthew Carson

Matthew Carson was born in Blacksburg, Virginia, on March 31, 1975. He lived in Columbia, South Carolina, until the age of 12, and moved to Blountville, Tennessee, in 1987. He graduated from Sullivan Central High School with honors in June 1993, and began attending Virginia Tech in August 1993. He graduated Summa Cum Laude from Virginia Tech with a Bachelor of Science in Electrical Engineering in May 1998. During his time at Virginia Tech, he participated in various clubs including the Baptist Student Union and Formula SAE. He also did four co-op terms with General Motors in Michigan. Upon graduation, he accepted a full-time job with General Motors in Milford, Michigan. Here he worked in Assembly Verification supporting software used to test cars at the end of assembly lines. He traveled to many assembly locations and earned valuable real world experience. In August 2000 he returned to Virginia Tech to pursue his Master of Science in Mechanical Engineering. His graduate research consisted mainly of analysis of pulsed control signals. In December 2001, he successfully defended his Master's thesis and graduated from Virginia Tech. He has accepted employment with Joe Gibbs Racing in Huntersville, North Carolina, and will be working in the Winston Cup engine shop supporting dyno equipment and performing engine development work.