Title

Efficient DC-DC switch regulator : applied iterative learning and anti-windup control

Author

Raley, Jack

Publication Date

2008

UNIVERSITY OF CALIFORNIA, SAN DIEGO

Efficient DC-DC Switch Regulator: Applied Iterative Learning and Anti-Windup Control

A thesis submitted in partial satisfaction of the
requirements for the degree
Master of Science

in

Engineering Sciences (Mechanical Engineering)

by

Jack Raley

Committee in charge:

Professor Raymond A. de Callafon, Chair
Professor Thomas R. Bewley
Professor Miroslav Krstić

2008

The thesis of Jack Raley is approved and it is acceptable in quality and form for publication on microfilm and electronically:

Chair

University of California, San Diego

2008

DEDICATION

To my parents,
Vance and Cathy

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

Variable	Units (SI)	Description
$y_d(t)$	volts	Desired Voltage Output Signal
$y_j(t)$	volts	Actual Voltage Output Signal
$y(t)$	volts	High Side Voltage Signal
$u(t)$	volts	Control Signal
$u_j(t)$	volts	Iterated Control Input Signal
$d(t)$	amps	Pulse Current Demand Signal
$d_k(t)$	amps	Periodic Pulse Current Demand Signal
$r(t)$	volts	Set Reference Voltage Value
$r_j(t)$	volts	Periodic Step Input Signal
$s(t)$	volts	Step Response Data
$z(t)$		Integral of Error Signal
$e_j(t)$	volts	Closed-Loop Error Signal
$G(z)$		Discrete Plant Transfer Function
$\hat{G}(z)$		Estimate of Discrete Time Plant Transfer Function
$G(q)$		Plant Filter
$C(z)$		Discrete Controller Transfer Function
$C(q)$		Controller Filter
$F(q)$		Filter Describing Ideal Feedforward Signal
$T(z)$		Discrete Closed-Loop Negative Feedback Transfer Function
$U_{K_I}(t)$		Integrator Control Effort
U_{max}, U_{min}		Bound Limits Set On Integrator
$K_P(z)$		Discrete Proportional Control
$K_I(z)$		Discrete Integral Control
$K_D(z)$		Discrete Derivative Control
$\hat{F}(z)$		Estimate of Discrete Time Closed-Loop System
$S(z)$		Discrete Time Sensitivity Function
L		ILC Learning Function
Q		Q-Filter
q		Forward Time Shift Operator
μ		Power Efficiency
P	watts	Power
V	volts	Voltage
R	Ohms	Resistance

ACKNOWLEDGEMENTS

I would like to thank Professor de Callafon for taking me on board and allowing me to work on this project. His time and patience throughout my graduate studies have given me a true appreciation for the dedication he has to the field. The system identification methods that he presented me were a key factor in beginning this project and proved likewise in its completion.

Charles Kinney has been a major motivation in my pursuit of this thesis project and a continuing source of inspiration.

David Szeto provided me with insight to the programming process and many other helpful tips throughout the project.

Brandon Wiedemeier and I had numerous discussions on approaches to the control problem at hand. Our logical progressions on issues concerning the project kept it in motion.

ABSTRACT OF THE THESIS

Efficient DC-DC Switch Regulator: Applied Iterative Learning and Anti-Windup Control

by

Jack Raley
Master of Science in Engineering Sciences (Mechanical Engineering)

University of California San Diego, 2008

Professor Raymond A. de Callafon, Chair

A system model of a small modern power regulator is developed and tested to analyze the possibility of increasing power efficiency. The model is constructed using MATLAB software, and its switch linear performance in the presence of periodic pulse loads is verified against that of the actual system. Two enhanced control techniques, Anti-Integral windup and Iterative Learning Control, are individually applied to the model and used to reduce the deviation of the systems high side voltage response level when under load. Finally, the combined implementation of the two methods is explored with results showing that a 12.5% increase in total power efficiency can be achieved along with a reduction in response settling time. Accuracy of the simulated system suggests that these improvements can be translated to the real system with the addition of the two enhanced control techniques tested. A typical DC-DC converter such as the one modeled after with 80–85% efficiency would benefit by increasing its efficiency up to the high 80–90% range. Here the improvement is from 80.4% efficiency to 92.9%.

Chapter 1

Introduction

With soaring energy costs quickly becoming a global concern, the need for efficient energy usage is on the rise. The once abundant outlook on fossil fuel supply is waning due to increased demand from a growing population and continued development of third world countries. Fuel prices for transportation have reached on all time high bringing stress and hardships to many global economies. Humans are trying now more than ever to reclaim wasted energy and build more efficient products and vehicles. The rise in a “green” way of thinking has sparked the publics interest in clean machines such as hybrid vehicles. These vehicles increase the efficiency of their fossil fuel use by converting lost energy during braking to electrical power used to keep it in motion. The recent demand for hybrid vehicles has been driven by the exponential increase in oil and gasoline prices over the past few years. Awareness of the scarcity of oil has driven up its price, and has also brought about awareness of dwindling supplies of other fossil fuels. Coal and natural gas for example are used in many large power plants to provide energy to cities throughout the world. These fossil fuels are also limited in their supply and the world may soon face the same increase in usage cost of them as it has with oil. Optimizing the efficiency of energy usage is critical to sustaining the world as it is today. With the advent of the digital age comes more opportunity for technology to improve upon outdated machinery and provide for an increase in power and energy efficiency. High speed computers and optimizing algorithms are readily available to be implemented on all types of power systems large and small. Such technological advancements can be applied in machinery and the power supplies that drive them, thereby becoming the core components to a more efficient and energy friendly future.

1.1 The SpikeSafe200 Power Supply

The SpikeSafe200 (SS200) shown in Figure 1.1, is a switched DC-DC current source power regulator manufactured by a small San Diego based electronics company. It is capable of handling up to 200 volts total with each one of its eight channels delivering up to 5 amps of current a piece. Vektrex, the manufacturer responsible for the SS200, provides custom power, testing, and measurement solutions to numerous companies in the Southern Californian region. While the aim of their custom builds is to be as efficient as possible, there always exists areas of development where further improvements can be made. This thesis addresses some areas where the SS200 lacks in efficiency and improves upon the overall performance of the power supply model.

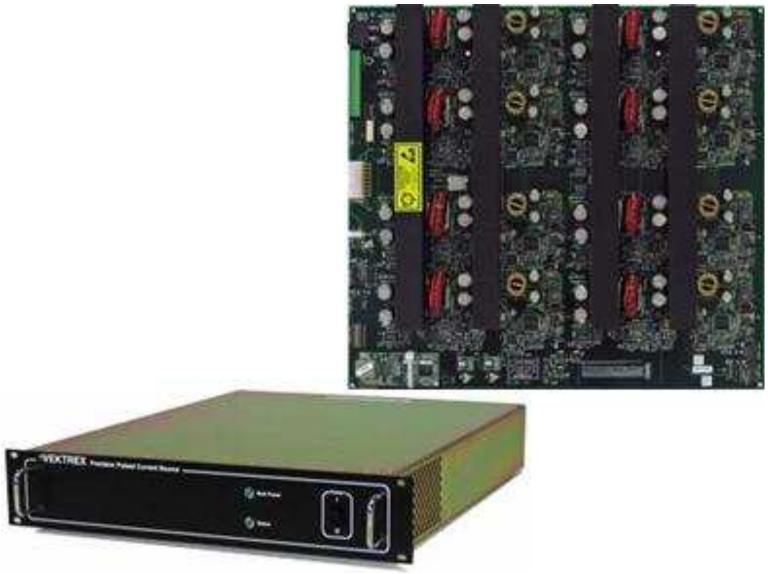

Figure 1.1: The Vektrex SpikeSafe200

1.2 Problem Statement

Most modern DC-DC regulators exhibit some sort of power loss in their performance which puts their range of efficiency anywhere from 80–85% [8]. Often times these ineffficencies stem from inadequate control of the regulator's supply voltage level when under sudden on/off loading. An example of this is given by the oscillatory behavior seen in Figure 1.2. This figure is a screen capture provided by Vektrex of the SS200 performance response under a periodic pulse load. The two areas of concern where energy is not being utilized optimally are pointed out in red. The green line in the screen capture is the power supply's high side voltage signal $y(t)$, which represents a set amount of voltage that is regulated by the power supply. Note that it is oscillating about a constant reference value, $r(t)$. The yellow line in the figure represents a pulse current demand $d(t)$, which has been placed on the system. In this particular test case the demand pulse occurs periodically every 16ms with an "ON" (high) time of 1ms. The system is stable in regards to the demand signal, but oscillations in the response occur after both the rising and falling edges of each new demand pulse. These oscillations are actually a result of the power supply's feedback control loop. The onboard second order PID controller $C(z)$, lacks enough freedom to fully and quickly compensate for the periodic demand during the given "ON" time frame. The "OFF" time portion which comprises the remainder of the demand period is somewhat governed in length by an additional constraint embedded in the power supply's circuitry. The controller returns the system voltage level to the desired reference value but must do so while under restrictions of its falling slew rate. Once the reference value has been settled upon, another pulse demand can be imposed on the system. To improve the efficiency of this power regulator it must be able to switch on and off as quickly as possible with as little overshoot or undershoot as possible. This means that the regulator's supply voltage level needs to be controlled in a very robust manner.

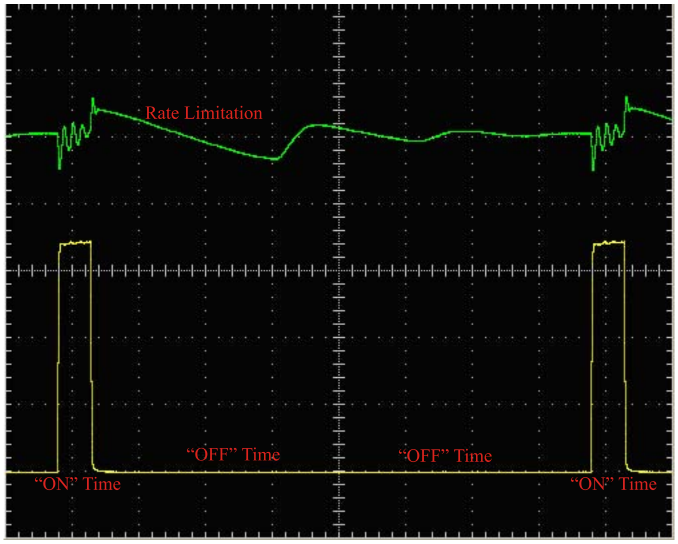

Figure 1.2: Vektrex SpikeSafe200 Performance Scope Shot

Advanced control methods capable of tightly regulating the supply voltage level are available, but must first be tested in order to demonstrate their ability of enhancing the system's performance. These tests must be run on an accurate model representation of the SS200 so that the performance improvements can be translated clearly. This requires that real experimental data from the SS200 be gathered and used to model the dynamics of the system. The new control methods can then be implemented to improve upon the model response by reducing both the transient oscillations that occur and its settling time. Increasing the actual SS200's efficiency begins with illustrating the effectiveness of additional advanced control methods.

This includes:

- Modeling the SS200 system dynamics
- Minimizing transient response oscillations
- Reducing response settling time

1.3 Objective

The main objective of this thesis is to demonstrate the increased efficiency that can be obtained in a DC-DC power regulator with the addition of advanced control algorithms. The SS200 system dynamics will be modeled from real experimental data to ensure the accuracy of its characteristics. This will be performed with both open and closed-loop experiments along with a few key identification techniques. The model will then be used as a platform to verify and demonstrate the improvements that are possible from implementing advanced control methods. Iterative learning control will be applied during the demand "ON" time in order to eliminate the transient oscillations that occur there. During the "OFF" portion of the reponse an Anti-Integral windup technique will be utilized to reduce the settling time of the system along with the transients that remain. The combined effectiveness of these two methods will ultimately be demonstrated as motivation for their development on the actual SS200 system.

1.4 Outline of Thesis

Modeling of the SS200 begins in Chapter 2. Closed-loop experiments to obtain real SS200 response data are discussed along with the identification of the system dynamics. The SS200 plant dynamics $G(z)$ are computed via open-loop identification and curve fitting methods. This plant model is then used with the known PID controller $C(z)$ to develop a SIMULINK model in MATLAB. Verification of the model is performed by comparison with the real experimental SS200 response data from Vektrex.

Chapter 3 identifies the issues associated with the "OFF" time transient oscillations. The characteristics of the problem area are discussed and the use of Anti-Integral windup control is motivated. Two applications of this control technique show success in its ability to reduce oscillations in the response as well as shorten the settling time as intended.

Chapter 4 targets the issue of transients durring the "ON" demand time frame. Here, *ideal* feedforward control is first introduced to convey the power behind this type of approach to the problem. However, the non-ideal nature of the SS200 dynamics give motivation for a more robust control method such as iterative learning control to be implemented. The results of applying this method are given after a brief discussion of its learning function which is the core component of the advanced technique.

Chapter 5 brings together the improvements of both control methods by giving a procedure for combining the two. The results of their dual implementation are discussed along with the overall improvement to the response of the system model.

Finally a comparison study is provided in Chapter 6 to discuss the improvements that can be made using the iterative learning control technique versus that of a simpler feedforward pulse signal. This comparison further demonstrates the power of the iterative learning control and its ability to completely reject a periodic disturbance even in the presence of model uncertainty.

Chapter 2

Modeling SS200 Open-Loop Dynamics

2.1 Introduction

All implementation and testing of the advanced control methods are performed on a model of the SS200 system. This makes the development of an accurate simulation of upmost importance. The model is needed in order to illustrate and analyze the effectiveness of the control solutions proposed. In addition, a model based simulation is the only valid way of implementing and testing these theories without proper knowledge of the power regulator's software programming. The purpose of this chapter is to explain how the model was created and verified as accurately representing the actual unit. The system model is built in MATLAB's simulation and model based design tool SIMULINK utilizing experimental response data taken from the SS200. The controller equation for the SS200 is known $C(z)$, however the plant equation $G(z)$ is not. This chapter begins with how performance data was obtained and used to estimate the mathematical representation of the plant dynamics. From there the two critical components of the system are placed in SIMULINK and the simulation model is completed.

2.2 Closed-Loop Experiments and Identification

Accurate representation of the SS200 system dynamics in the model are of extreme importance for this study. The precision of model characteristics will ensure that the enhanced control techniques implemented here are not performed in vain. To provide for the most reliable and up to date information, closed-loop experiments are performed on the SS200 unit. The data gathered from these experiments is then used in system identification techniques to best approximate a mathematical representation of the closed-loop system. From this representation and prior knowledge of the PID controller $C(z)$ currently in the SS200, the actual plant dynamics can be extracted for use in the MATLAB model.

$$C(z) = \frac{15z^2 - 29.73z + 14.74}{z^2 - 1.942z + 0.942} \tag{2.1}$$

2.2.1 Data Collection

In order to collect data from our SS200 unit and also mimic the types of demand signals seen by the power supply, step input signal's were tested. The block diagram seen in Figure 2.1 illustrates the basic setup of these experiments. The closed-loop SS200 system to be identified is represented by the discrete time transfer function $T(z)$.

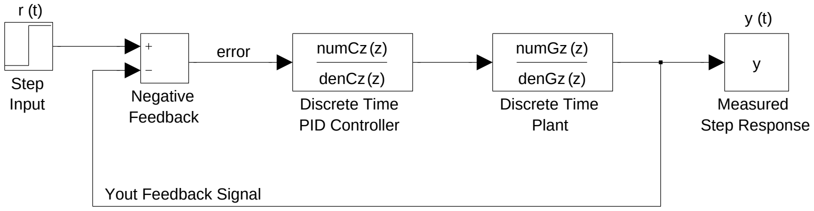

Figure 2.1: Data Collection Configuration

$$T(z) = \frac{G(z)C(z)}{1 + G(z)C(z)} \tag{2.2}$$

This equation represents the negative-feedback series connection of the known controller $C(z)$ and the unknown plant $G(z)$ which is to be estimated. It is well known that a DC-DC switch power supply can be modeled by a negative-feedback loop [14] [4]. The periodic step inputs $r_j(t)$ are configured to represent a known load configuration often placed on the system.

$$r_j(t) = \begin{cases} 1.2, & 0 \le t \le 0.005s, \\ 0, & 0.005 < t \le 0.015s \end{cases} \tag{2.3}$$

with j the 15ms period index.

This step input mimics the effects of a 3A 20 diode configuration, and is used to excite and extract the full response profile from the systems output signal $y(t)$. Note that the Vektrex power supply is intended for the use of testing long strings of LEDs in a similar pulsed fashion. The data collected from the experimental tests is analzyed and one portion of the step response is singled out in particular. This singled out portion is defined as $s(t)$ and represents a duration of time where the input to the system is pulse like in structure and fully excites a linear repsonse from the system. Figure 2.2 below displays the system response data measured with the 3A 20 diode configuration, the singled out step portion $s(t)$ is highlighted in red.

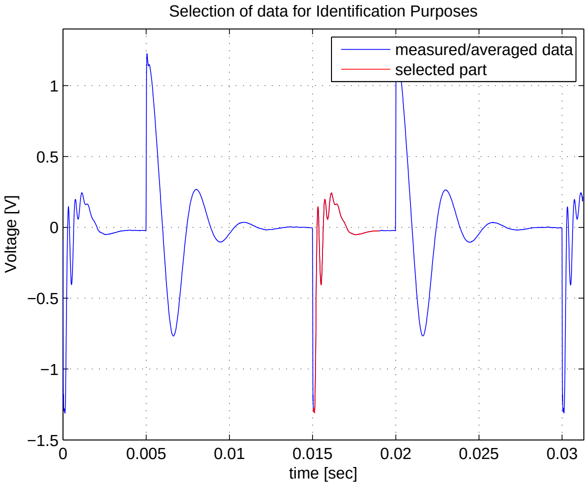

Figure 2.2: SS200 Measured Step Response

2.2.2 Closed and Open-Loop Identification

Due to the inherent switched nature of the regulator system and the experiments performed, our data consists of step responses. The data set $s(t)$ consists of 524 samples. The step realization algorithm of Callafon et al. 2003 [5] is used to formulate the closed-loop system model $T(z)$ directly on the basis of the measured step response data. This method is a generalization of the conventional Hankel matrix based realization algorithm [6]. For more details on the identification method used here one is referred to [5].

To begin, we will determine the frequency response of the SS200 closed-loop system by first identifying it as a linear discrete time state space realization

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \tag{2.4}$$

which can be rewritten in the form [6]

$$y(t) = Du(t) + \sum_{j=1}^{\infty} G(j)u(t-j), \quad G(j) := CA^{j-1}B \tag{2.5}$$

with Markov parameters $G(j)$, $\triangle$T normalized to 1, and time t measured in discrete samples k for $k = 0, 1, \ldots, N$. This gives a discrete time input-ouput relationship that, following [6], can be rewritten in the Hankel based representation

$$Y = HU + E \tag{2.6}$$

where H is the N_1 x N_2 Hankel matrix of step coefficients, U is a square non-singular N_2 x N_2 Toeplitz matrix storing the input data, and E is a N_1 x N_2 matrix containing the effects of past input signals $r(t)$ multiplied by the Markov coefficients $g(k)$. Recall from above that $N = 524$, and $N_1 + N_2 \leq N$ with $N_1 = N_2 = 262$. This relation can then be rewritten as

$$Y - E = HU$$

where we can define R to be the input/output matrix $R := Y - E$. From the singular value decomposition (SVD)

$$R = U\Sigma V^T = [U_n \; U_s] \begin{bmatrix} \Sigma_n & 0 \\ 0 & \Sigma_s \end{bmatrix} \begin{bmatrix} V_n^T \\ V_s^T \end{bmatrix} \tag{2.7}$$

we can approximate R by the rank n matrix R_n given by

$$R_n = R_1 R_2$$

where

$$\begin{aligned} R_1 &= U_n \Sigma_n^{1/2} \\ R_2 &= \Sigma_n^{1/2} V_n^T \end{aligned}$$

with rank determined from SVD analysis as $n = 12$. The shift property of the Hankel matrix extends to the defined R matrix such that the shifted matrix $\bar{R}$ holds the similar relation

$$\bar{R} = R_1 A R_2$$

From here it can be shown that the state space realization is given by

$$\hat{D} = s(0),\ \hat{C} = R_1(1,:),\ \hat{B} = R_2(:,1),\ and\ \hat{A} = R_1^{\dagger}\bar{R}R_2^{\dagger} \tag{2.8}$$

of order 12 with the left inverse of R_1 and the right inverse of R_2 given by

$$R_1^{\dagger} = \Sigma^{-1/2}U_n^T$$
$$R_2^{\dagger} = V_n\Sigma_n^{-1/2}$$

respectively. The state-space model of the closed-loop system is simulated with a step input to compare and verify its response with the measured experimental step response data. Figure 2.3 shows this comparison and is a clear verification that the identified state-space model accurately characterizes the dynamics of the SS200 closed-loop system. The step response of the model fits nearly perfectly over the experimental step response data obtained at Vektrex. The frequency response of this model $T(e^{j\omega\Delta t})$ is captured in Figure 2.4. Data from this frequency response is used with knowledge of the controller[1] $C(e^{j\omega\Delta t})$, to extract the plant frequency response data, $G(e^{j\omega\Delta t})$. This is perfomed via cancellation of the controller dynamics from the identified closed-loop dynamics seen in Figure 2.4. Calculations done in this order allows one to curve fit a low

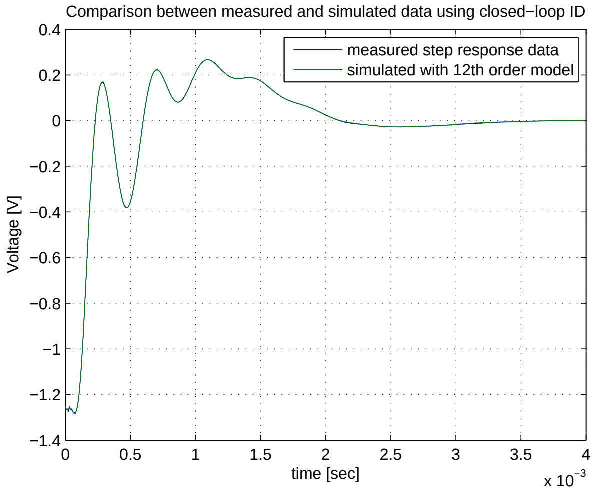

Figure 2.3: 12^{th} Order Modeled CL Response vs. Measured CL Response

[1] The one-sided z-transform of the signal $\{x(k)\}_{k=0}^{\infty}$ is $X(z) = \sum_{k=0}^{\infty} x(k)z^{-1}$, with the z-transformation of a system being obtained by replacing q with z. The frequency response of a z-domain system is given by replacing z with $e^{i\theta}$ for $\theta \in [\pi, -\pi]$,[3]

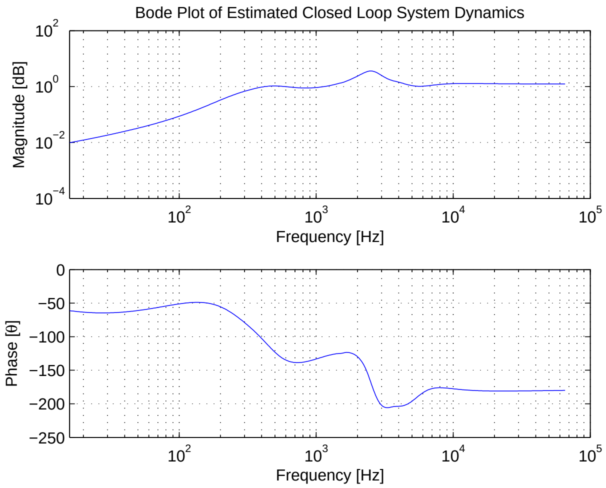

Figure 2.4: 12^{th} Order Closed Loop Model Bode Response

order, linear, discrete time model to the plant frequency response data. This technique is explained in [7]. The resulting open-loop frequency response of the plant is displayed in Figure 2.5. A Least Squares (LS) curve fitting optimization is performed with a theoretical order magnitude of 6. The reader is referred to Chapter 4 and [2] for more details on LS curve fitting. The frequency response of the fitted plant model $G(z)$ is then compared with the computed frequency domain representation in Figure 2.6 to ensure its accuracy. Verification of the fitted plant model is given with the two responses lining up

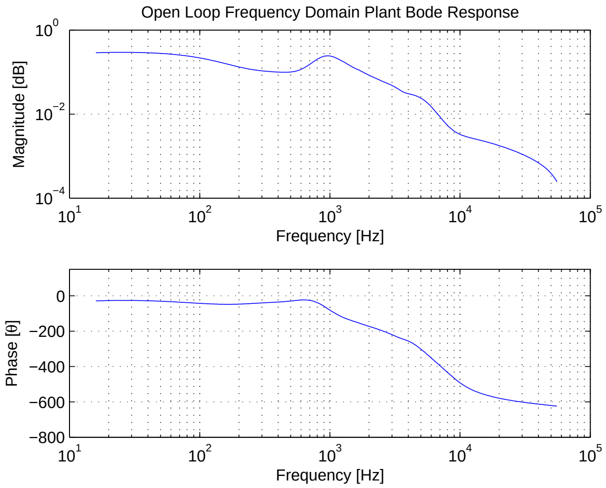

Figure 2.5: Frequency Domain Open Loop Plant Bode Response

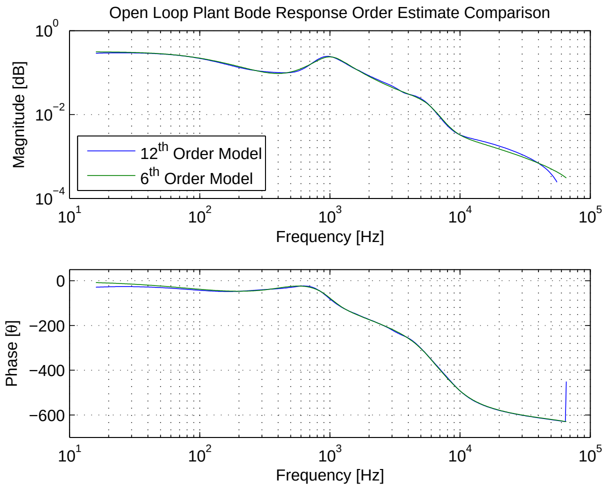

Figure 2.6: Frequency Domain Open Loop Plant Bode Response

on top of each other. This indicates that the plant identified $G(z)$, represents the real plant accurately. The fitted LS equation for $G(z)$ is given in (2.9) with corresponding coefficients in Table 2.1.

$$G(z) = \frac{b_0z^6 + b_1z^5 + b_2z^4 + b_3z^3 + b_4z^2 + b_5z + b_6}{a_0z^6 + a_1z^5 + a_2z^4 + a_3z^3 + a_4z^2 + a_5z + a_6} \tag{2.9}$$

The step response of the closed-loop negative-feedback system $T(z)$ comprised of $C(z)$ and $G(z)$ is then compared with the step response of the 12^{th} order state-space realization (2.8) in Figure 2.7. The estimated system response tracks the 12^{th} order system response almost perfectly. There are some slight differences in oscillation but overall the two curves have a tight fit. A second comparison is analyzed between the bode response plots of the two systems, Figure 2.8. Here again the two curves are lined up showing only a slight deviation in their phases at lower frequencies. These comparisons show that the model obtained for the SS200 plant are valid and accurate.

Table 2.1: Estimated 6^{th} Order Plant Coefficients

Numerator	Denominator
b_0 = -0.0003431	a_0 = 1
b_1 = 0.001395	a_1 = -5.761
b_2 = -0.001631	a_2 = 13.85
b_3 = -0.000632	a_3 = -17.79
b_4 = 0.002894	a_4 = 12.88
b_5 = -0.00228	a_5 = -4.983
b_6 = 0.0005984	a_6 = 0.8047

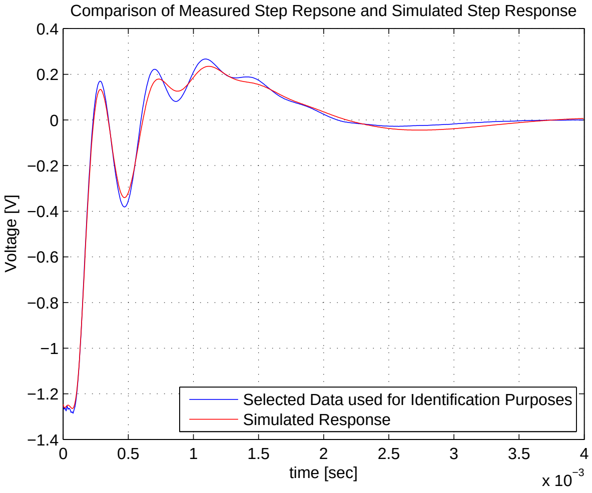

Figure 2.7: 6^{th} Order Plant and Controller CL System Step Response Compared to Measured Step Response

2.3 Model Structure and Verification

The SIMULINK block model begins with a standard unity negative-feedback loop and the discrete time controller (2.1) and plant (2.9). The high-side voltage performance response $y(t)$ of the SS200 is the result of a periodic pulse demand $d_k(t)$ being placed on the system. The power supply maintains a set voltage level which becomes disrupted when the demand signal enters the system. Similarly in the block model, the system is set to maintain a constant reference voltage $r(t) = 0$, and becomes disrupted by a periodic demand disturbance. The periodic pulse disturbance in the model is injected into the system at the plants output. This orientation produces a simulated response which is very similar to that of the SS200's response during the "ON" period. The period duration and "ON" time of the disturbance pulse is the same as that of the demand pulse observed in the data collection experiments, 15ms and 5ms respectively. The response of the system with the arrangement as described is given in Figure 2.10. Comparison to the actual system response shows that the "OFF" time portion of the model response is lacking in form. The "OFF" time characteristics of the SS200 are non-linear and exhibit a dampening slow oscillation as compared with the "ON" time. The non-linear effects that slow this oscillation are caused by a rate limitation on the system. Designed in the power supply's circuitry is a resistor that bleeds off excess energy at a limited rate. This rate limitation inhibits the system's response and causes the elongated settling

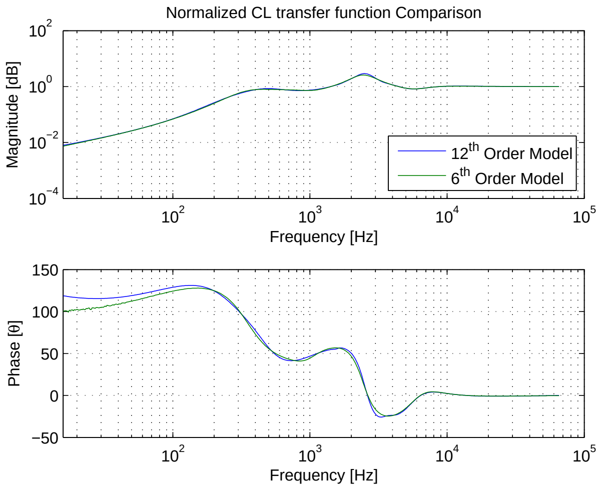

Figure 2.8: Fitted CL System Bode Response Compared to 12^{th} Order CL System Bode Response

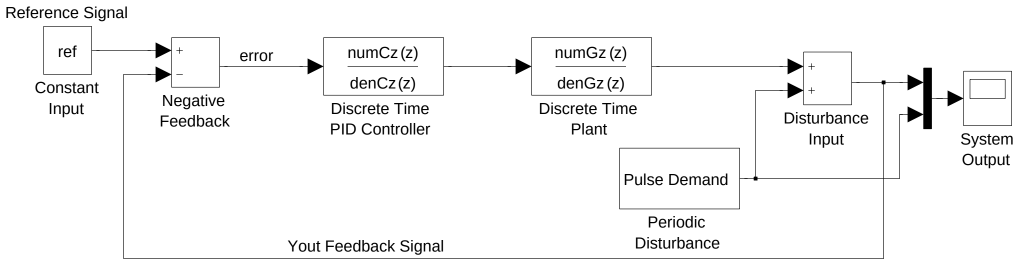

Figure 2.9: First Stage SIMULINK Model

time. The rate limitation only occurs during the pulse "OFF" time making the SS200 a switch linear/non-linear system. *A priori* knowledge of this rate limitation leads to the addition of a rate limiter in the SIMULINK model. The switch in the SIMULINK model is placed at the plant output after the rate limiter to mimic this behavior. It is triggered by the pulse disturbance to allow either a normal or rate limited signal through the system. Figure 2.11 shows the SIMULINK model with the additional rate limiter. The response comparison in Figure 2.12 clearly demonstrates similar characteristics of the modeled system dynamics to those of the actual system. At the rising edge of the periodic disturbance there is an initial dip in the system output followed by dampening oscillations. At the falling edge of the disturbance there is a slight spike followed by dampening slow oscillations. The rate limitation placed on the SIMULINK model has been set from rate calculations observed in the example Vektrex response plot. This fi-

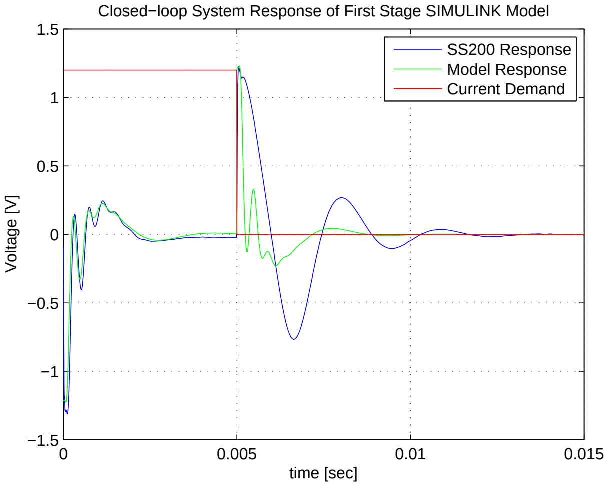

Figure 2.10: Purely Linear SIMULINK Model Response

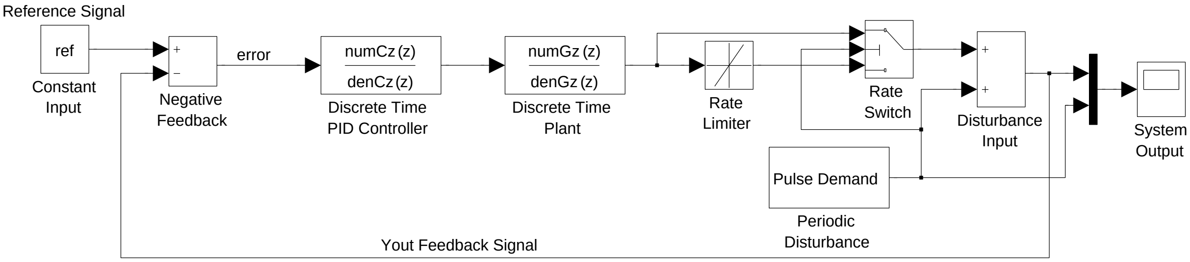

Figure 2.11: Negative Feedback Control Loop with Switched Rate Limitation

nal comparsion marks the completion of the modeling and verification procedure. It has been shown that the SS200 system dynamics can be modeled by the negative-feedback connection of an identified linear 6^{th} order plant model in series with the known PID controller, along with a saturating element which models the effect of output saturation embedded in the units circuitry.

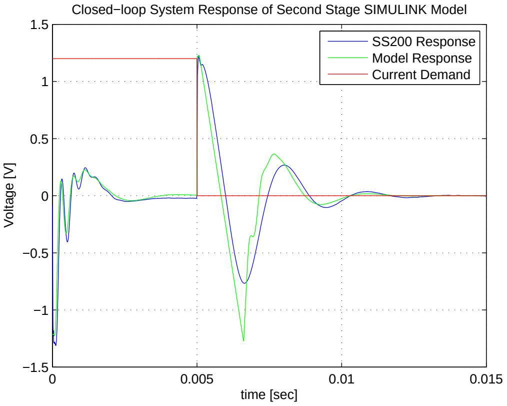

Figure 2.12: SS200 Response vs. Modeled Response

Chapter 3

Eliminating Transient Oscillation with Anti-Windup Control

3.1 Introduction

The SS200 system exhibits two distinct problem areas with response characteristics slightly differening in each. The switch linear/non-linear behavior that occurs in these respective regimes will be dealt with using two separate enhanced control techniques. The first of these two areas occurs during the pulse "OFF" time where the falling slew rate limitation on the system causes slow oscillations and a longer settling time. The controller itself is not aware of the performance limits of the system and continually increases its efforts as a result of the prolonged error signal it recieves from the feedback loop. Soon after the controllers response becomes saturated and the non-linear behavior that is observed during the disturbance "OFF" time begins. It is the integrator portion of the PID controller that increases in value despite exceeding the point of its relative usefullness. This phenomenon is better known as integral windup [17].

Integral windup refers to the value of the integral portion of a PI or PID controller reaching its maximum or minimum effective value and then continuing beyond that value. When the control efforts have finally restored the system output to the desired reference point the integrator cannot react instantaneously to stop providing a control signal. The integrator must first unwind from its over exerted value which causes a delay in the response of the system. This delayed response from unwinding produces longer and more pronounced oscillations [16], which is exactly the observation that can be made in Figure 1.2. The saturation of the modeled response leads to integral windup that in turn causes further oscillations and a longer settling time of the system.

3.2 Anti-Windup Theory

To combat against integral windup one can effectively force the integrator to be aware of its limitations by applying Anti-Windup logic to it. This logic is applied to the error signal before it passes to the integrator to evaluate whether or not the integrators control efforts are necessary. Anti-Windup logic allows the integral control effort $u_{K_I}(t)$ to function normally within a given set of bounds (U_{max},U_{min}), and limits this effort when the system's error drives the control signal to exceed these bounds [13].

$$\begin{gathered} U_{max} \text{ if } u_{K_I}(t) \geq U_{max} \\ u_{K_I}(t) \text{ if } U_{min} < u_{K_I}(t) < U_{max} \\ U_{min} \text{ if } u_{K_I}(t) \leq U_{min} \end{gathered}$$

Essentially, it provides the integrator with knowledge of its saturation constraints and its current state relative to those constraints. By "shutting off" the error signal to the integrator when it has reached its maximum or minimum value the Anti-Windup logic prevents the controller from marching beyond a useful magnitude. This is also known as *conditional integration* or *integrator clamping* [17]. The controller can then react more quickly to the system response returning to or surpassing the reference value in the opposite direction. A formal explanation of this concept is provided below.

Let the system's output error be defined as $\dot{z}(t) = r(t) - y(t)$, such that $z(t)$ is the integral of the output error signal over some time period $[0, t]$. The desired control signal of the PID controller becomes

$$u_{des}(t) = K_P \dot{z}(t) + K_I z(t) + K_D \ddot{z}(t)$$

But in order to apply the Anti-Windup logic to the integrator it must first be distinguished from its proportional and derivative counterparts. Pulling out the integral portion of the PID controller yields

$$u_{K_I}(t) = K_I z(t)$$

Anti-Windup bounds are initially set to some U_{max} and U_{min}. Prior knowledge of the amount of control effort available from the controller will help to provide insight on where to first set these bound limits. Without such knowledge, it is up to the control designer to set a generous limit in the Anti-Windup logic and observe the simulated response. The bounds can be decreased iteratively until a desired performance from the system is achieved. In this application the continued decrease in bound limits eventually resulted in an adverse response of the simulation. A severely strict limit on the integral control effort will prevent the system response from ever returning to the reference level [17]. Applying Anti-Windup logic to the integrator takes the following form [13]:

$$\begin{aligned} if\ u_{K_I}(t) &\geq U_{max}\ and\ r(t) \geq y(t) \Rightarrow z(t) = 0 \\ elseif\ u_{K_I}(t) &\geq U_{max}\ and\ r(t) \leq y(t) \Rightarrow z(t) = \int_0^t [r(t) - y(t)]dt \\ if\ u_{K_I}(t) &\leq U_{min}\ and\ r(t) \leq y(t) \Rightarrow z(t) = 0 \\ elseif\ u_{K_I}(t) &\leq U_{min}\ and\ r(t) \geq y(t) \Rightarrow z(t) = \int_0^t [r(t) - y(t))]dt \\ else\ u_{K_I}(t) &= K_I z(t) \end{aligned} \tag{3.1}$$

Where $u_{K_I}(t)$ again is the integral control output, $r(t)$ is the reference signal, and $y(t)$ is the system's output signal. The logic output $z(t) = 0$ sends an error signal of zero to the integrator preventing it from increasing in magnitude.

3.3 Implementation

Implementing the Anti-Windup scheme in the SIMULINK model first requires that some slight modifications to its configuration be made. The second order PID controller must first be decomposed into its three distinctive parts: Proportional action, Integral action, and Derivative action. This is accomplished by performing a partial fraction expansion on the controllers discrete time transfer function equation [9]. The resulting individual components from this procedure are given below.

Proportional:

$$K_P(z) = 15 \tag{3.2}$$

Integral:

$$K_I(z) = \frac{0.2179}{z - 1} \tag{3.3}$$

Derivative:

$$K_D(z) = \frac{-0.8172}{z - 0.9420} \tag{3.4}$$

Now that the three components are separated and available for individual manipulation they are put in place of the discrete time control block as seen in Figure 3.1. The error signal from the feedback loop is directed into to each component individually and their combined outputs are added together before passing on to the plant. Application of the Anti-Windup logic on the integrator term (3.3) is performed with an embedded function block placed in the SIMULINK model. The function block takes in the reference signal, system output signal, and integrator output signals which are called for in (3.1). With this information it can then calculate the appropriate input signal to send to the integrator. Since the rate limitation in this model only occurs during the "OFF" portion of the pulse disturbance a switch is placed after the embedded function to determine when the Anti-Windup logic is applied. The switch allows for an unmodified error signal to pass into the integrator when the pulse is "ON" and a modified error signal to pass into the integrator when the pulse is "OFF".

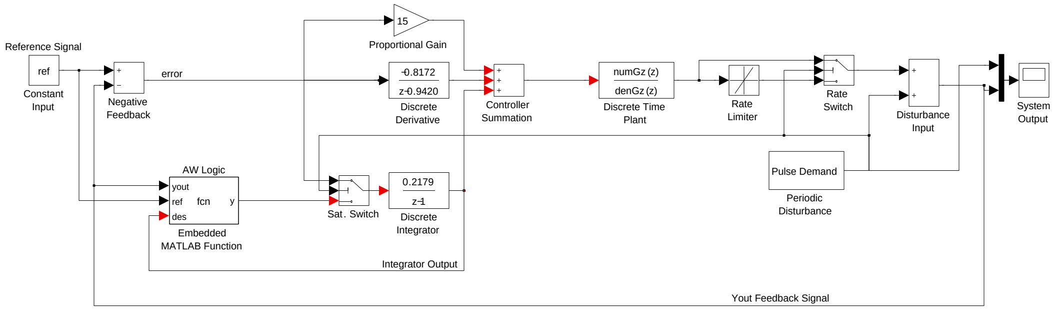

Figure 3.1: SIMULINK Model with Addition of Anti-Windup

3.4 Results

Performance results with the applied Anti-Windup scheme can vary as mentioned previously depending on the U_{max} and U_{min} values which are specified in the function block. Tuning these values can be done on a trial and error basis depending on the given system characteristics. Generally speaking, stricter bounds will result in less overshoot (or undershoot), fewer oscillations, and a shorter settling time. A consequence to stricter bounds maybe a longer rise time which results from the lower amount of control effort applied to the system. Conversely, more generous bounds will provide for a faster rise time with more overshoot and resulting oscillations. The goal of tuning the Anti-Windup limits is to find the values that work best for the particular system the logic is being applied to. If the desire is for no over/undershoot with less weighting on settling time then a stricter bounds is preferred, whereas if the desire is for a quicker settling time with little weighting on output oscillations a less stringent bound is used. To illustrate this point, two different bounds are applied to the modeled system and discussed here.

In order to anaylyze the improvements that can be gained with Anti-Windup control they must be measured against the baseline values of the model response with only PID control. The criterion for this comparison consists of the time the system response takes to settle within a 2% voltage range of the reference value, and also more importantly the increase in power efficiency. The regulators energy use over time is calculated with the equation for power given here:

$$P = \frac{1}{T}\int_0^T \frac{V^2}{R}\,dt$$

For these tests the voltage reference value is set at 5 volts with a demand disturbance of 2.5 amps. The "OFF" time length T is 10ms, sampling time dt is $5e^{-6}$s, and resistance R is 2 Ohms. The total period time is set to 15ms. Figure 3.2 displays the baseline system response of the model with this configuration. Note that the demand signal has been shifted up for ease of reference in the plot. The 2% settling time of the response is 6.1ms. Given a perfect regulator with no energy losses the power rate of the system would be 12.5 watts. However, the computed power rate for the modeled system comes out to be 16.49 watts during the "OFF" time frame. This means that the SS200 model

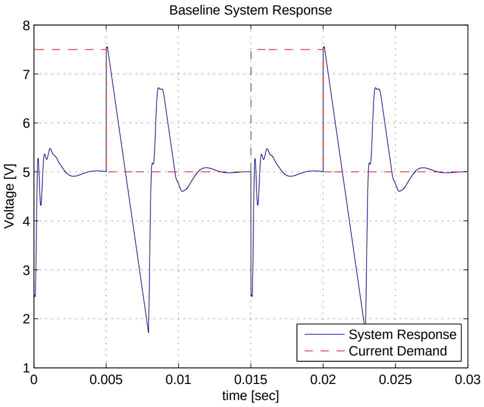

Figure 3.2: Model Response with 5 Volt Reference Signal without Anti-Windup Control

operates nearly 4 watts higher than what is demanded. The efficiency equation for power [15] gives the model an efficiency rating of 75.81% during this operation time.

$$\text{Power efficiency} = \eta = \frac{Output\, power}{Input\, power}$$

The system takes in power at 16.49 watts in order to output power at 12.5 watts.

While the Anti-Windup limits can be set up as high as infinity or as low as negative infinity, this would do the system no help in regulating the control energy put into the system. Conversely, if the limits are set too strictly then the system will be inhibited and fail to return to its set reference value. Inspection of the control signals produced by the SIMULINK model reveal that the control efforts of the integrator reach an upper limit no greater than 70 volts and a lower limit no less than -55 volts. This information is used to begin testing Anti-Windup bounds within the specified range. It should be noted that the system is only subject to constraints in its falling slew rate such that there is no needed bound limit for U_{max}.

3.4.1 Anti-Windup Limit: -20 V

Applying a $U_{min} = -20$ volt limit bounds with the Anti-Windup logic is a good start considering the -55 volt control effort observed from the integrator. At first glance of Figure 3.3 it may not appear as if the Anti-Windup logic has had a significant effect, but further analysis of the performance criterion proves otherwise. The system's settling time is reduced to 5.5ms, and there is a slight difference in the transient oscillations that occur. The reduction in the undershoot dip value and oscillations that follow result in a power rating of 15.03 watts, which translates to an increase in efficiency to 83.31%. This is a 7.51% improvement from the baseline rating. This improvement in efficiency of the response really exemplifies the negative impact that the integral windup was having on the system model.

3.4.2 Anti-Windup Limit: -1 V

For an even tighter system performance, a stricter bounds limit is placed on the system's controller. Here with the bounds lowered to $U_{min} = -1$ volts, the "OFF" time dip is decreased further and the settling time is improved to only 3.8ms. The smaller oscillations that result mean that even less energy is being bleed off from the system. The improved power rating for the "OFF" time duration is 14.14 watts. The Anti-Windup

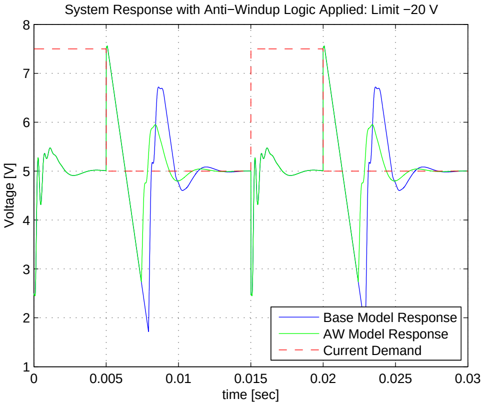

Figure 3.3: System Response with −20 V Anti-Integral Windup Limit

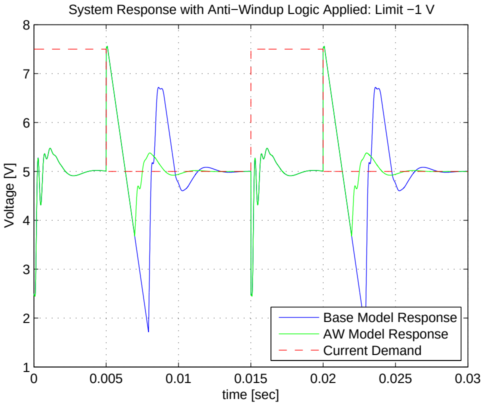

Figure 3.4: System Response with −1 V Anti-Integral Windup Limit

logic's further restraint on the itegrator output has increased the efficiency of the system by an astonishing 12.58% to 88.38%.

These tests results demonstrate that the Anti-Windup scheme performs as intended with significant improvements made to the response performance of the system model. The efficiency of the system is increased along with shortening of the response settling time. Through the use of *conditional integration* some of the system's poor performance due to a nonlinear saturation is able to be remedied.

Chapter 4

Iterative Learning Control

4.1 Introduction

The second of the two problems areas addressed occurs during the pulse “ON” time.

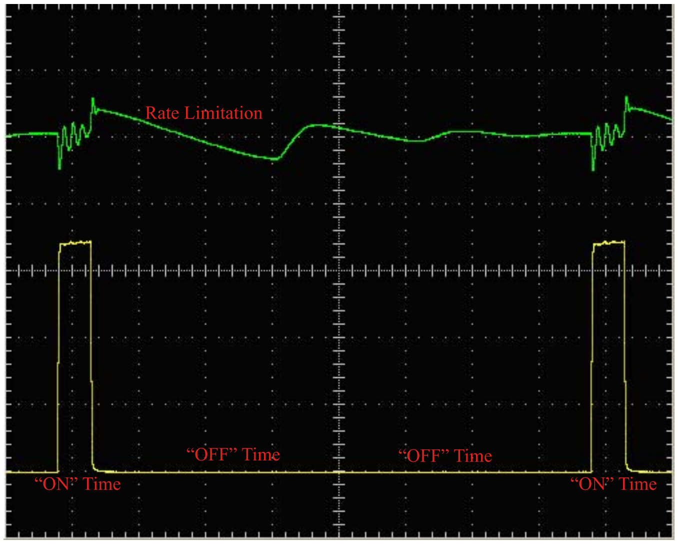

Figure 4.1: Vektrex SpikeSafe200 Performance Scope Shot

The oscillations observed here are due to a lack of freedom in the second order PID controller. The controller attempts to compensate for the pulse disturbance that suddenly enters the system, but it is not powerful enough to fully correct for the disturbance and bring the system back to the reference value in such a short time frame. These oscillations represent an area of inefficiency in the power supply’s performance that can be improved upon. A few important details that are known to the operator during this "ON" period are that the system is stable and that it behaves in a linear fashion. Additionally, the period and magnitude of the disturbance are known along with the length of its "ON" time. Insightful knowledge such as this, which is not often available, can be used in favor of the control system designer to implement additional feedforward control.

4.2 Ideal Feedforward Control

Feedforward control requires *a priori* or measured knowledge of the disturbance signal before it enters into the system. Knowledge of this disturbance is used to determine the impact it will have on the system and to calculate the control action needed to counteract that impact. In the ideal case, the user also has perfect knowledge of the controller and plant dynamics which allows for *ideal* feedforward control. To demonstrate the effectiveness of an ideal case a standard linear negative feedback loop is set up with the addition of a feedforward injection block. The ideal system described is illustrated in Figure 4.2 using the discrete time controller $C(z)$ and plant model $G(z)$ of the SS200 along with the known periodic demand signal $d_k(t)$; note that this system is stable under feedback control and does not contain rate limitations.

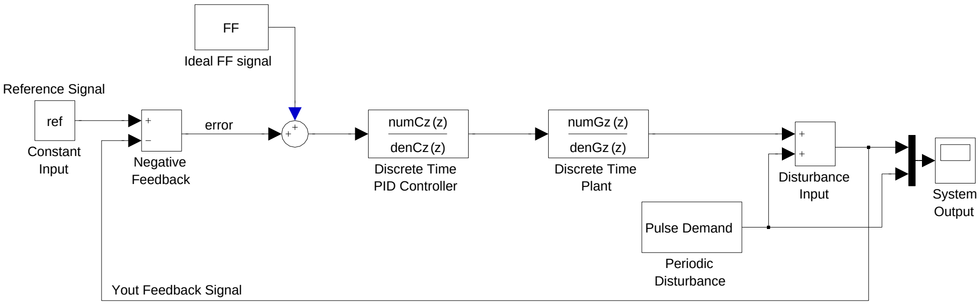

Figure 4.2: Negative Feedback loop with FeedForward signal Injection

The optimal *ideal* feedforward control requires perfect knowledge of the plant dynamics $G(z)$ as well as the controller dynamics $C(z)$, which is the case in this situation. If the disturbance signal is also known *a priori* then the ideal feedforward signal can be calculated before the system is ever put into motion. By performing these calculations on the periodic disturbance signal in the frequency domain using the fourier transform a non-causal feedforward signal is produced; this enhances the overall control of the system for future iterations. One very important detail to note is that the conditions of the system for which the feedforward signal has been calculated must remain the same for it to perform as intended. Analysis of the relationship between the disturbance input and feedforward input points in Figure 4.2 leads to the ideal feedforward signal calculation that follows. The calculations for the ideal feedforward signal are described in the time domain, which can also be performed in a non-causal manner given that the disturbance signal is known and does not change.

Creating a feedforward signal that is opposite in sign of the known disturbance signal through the inverse of the plant and controller dynamics can be accomplished in a non-causal manner by processing the disturbance data in both the forward and reverse directions. The filter required in this case is given below:

$$F(q) = \frac{1}{G(q)C(q)} \tag{4.1}$$

The resulting signal has zero phase distortion and will pass through the system from the entry point in Figure 4.2 and effectively cancel out with the plant and controller dynamics as it passes through. The remainder of this signal which is the opposite in sign equivalent of the disturbance signal will add together with the actual disturbance signal at its point of entry and equate to the reference value (zero).

It should be noted that the length of the feedforward signal is equivalent to the periodic length of the disturbance signal, and that a continuous looped repitition of both signals will yield a system output response with zero error. This can be seen in Figure 4.3 where the calculated *ideal* feedforward signal is displayed with a period length of 15ms. Note that due to the linear nature of this system the feedforward signal characteristics are the same, and only differ in the direction of their magnitudes depending on the demand pulse edge rising or falling. The simulated ideal system response before implementation of the *ideal* feedforward signal is seen in Figure 4.5; the blue line is the normal response of the system due to the known periodic disturbance under feedback control only. Notice how the system remains stable throughout the “ON” demand time and settles to the zero reference value during the pulse “OFF” time. Applying the ideal feedforward control signal (4.1) to the system in addition to its feedback control results in the near complete elimination of the output error. These results are seen in Figure 4.5 with the blue line holding a steady reference value of 5 volts. Again, this is an *ideal* situation in which all aspects of the system are known perfectly. In real world applications however, this is never the case. Often times the dynamics of a system may not be entirely accurate, and

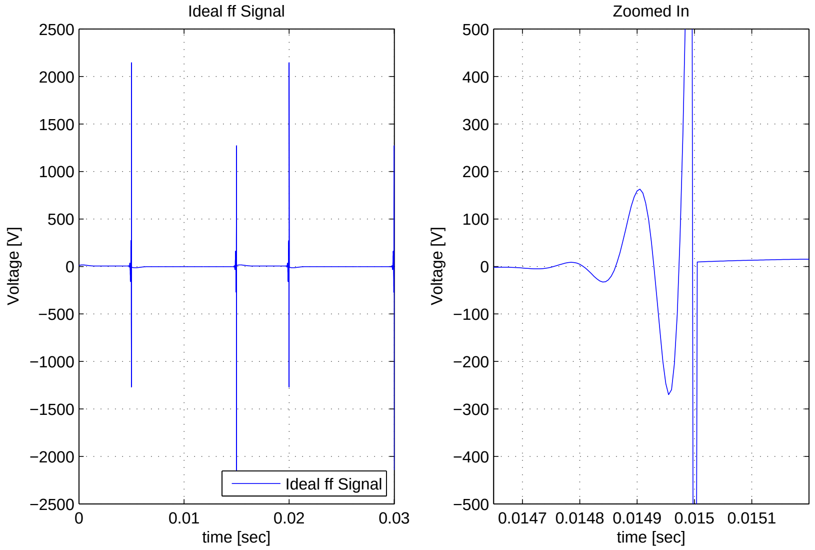

Figure 4.3: Repeating Ideal FeedForward Signal

many systems of interest cannot be completely characterized as purely linear. Systems such as these may in fact contain non-linear characteristics as well as linear ones. The SS200 in particular exhibits these switch system dynamics where both linear and non-linear behavior is observed. For improved control in this *non-ideal* case one can shift from an *ideal* feedforward solution to an advanced Iterative Learning solution.

4.3 Iterative Learning Control

Iterative learning control (ILC) is a notion that most humans have been familiar with their entire lives. The concept is one of generating open loop control from repetition and learning [3]. A prime example of this behavior takes the shape of a basketball player repeatedly shooting a basketball from the same spot over and over. As he improves his shot from the last repetition to the next he will learn and retain the memory of his successful shooting form [3]. This engrained memory will allow for more successful shots in the future, (i.e. less error). Similarly with ILC, a system under the same repeated task can learn from previous error measurements and correct itself to improve upon its future output performance. More specifically, this system generates a feedforward signal that will either improve upon the accuracy of a repeated desired reference signal, or improve its ability to reject a repeated disturbance signal. The power behind ILC comes

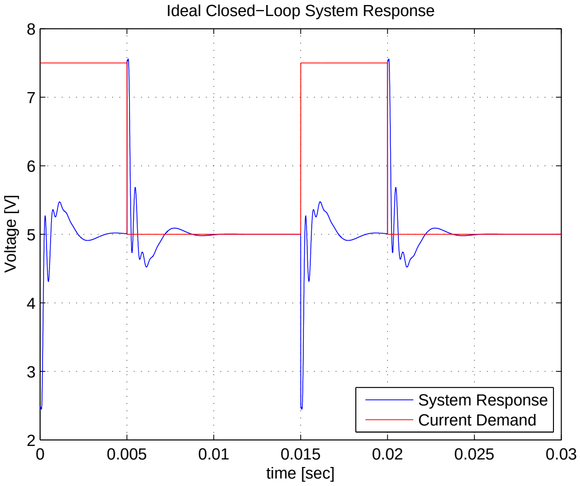

Figure 4.4: Ideal Linear System Response to Periodic Pulse Disturbance

in its ability to progressively learn and improve upon its performance even in the face of model uncertainties. In this instance we are using ILC as "an approach for improving the transient performance of [a] system(s) that operates repetitively over a fixed time interval." [1]

There are two conditions however, that a complete system must satisfy in order for ILC to be applied. These conditions are as follows [1]:

1. A tracked reference signal or rejected disturbance signal must have a uniform periodic behavior.

2. Initial Conditions of said system will remain approximately the same at the onset of each of these periods.

Slight deviations from these conditions can be tolerated under some ILC techniques but the resulting performance will degrade as the lack of adherence to them increases.

Introduction to ILC can best be done with a simple block diagram depicting the basic idea of what the controller is doing. The iterative learning controller requires measurements of the control signal to the plant u_j, the output of the plant y_j, and the desired output of the system y_d. The control input is then updated for the next iteration by filtering the systems error signal through some learning function L, and adding it to the previous control signal sent to the plant. With each iteration the control signal u_{j+1} is updated until finally no error is left and the future control signals converge.

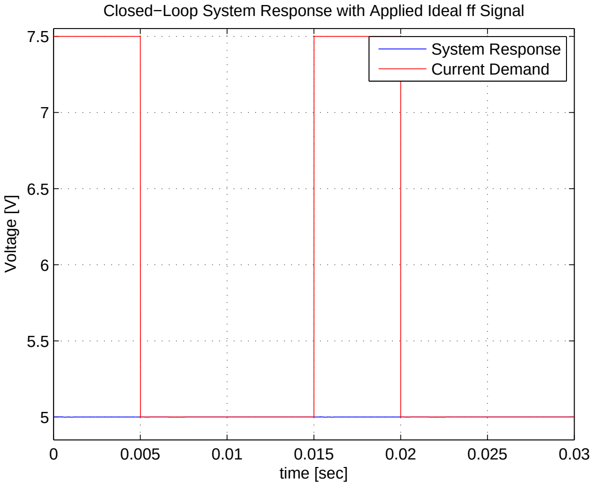

Figure 4.5: System Response with Implementation of Ideal FeedForward Signal

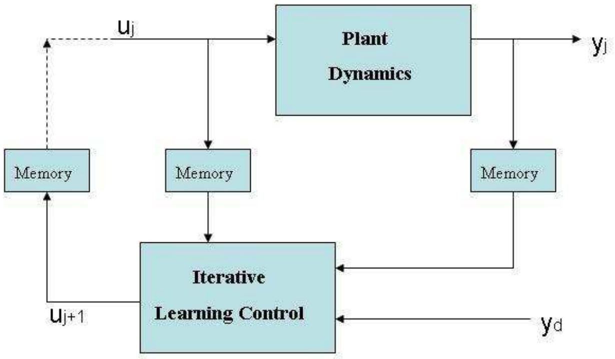

Figure 4.6: Block Diagram of ILC Concept [1]

Mathematically this looks like,

$$u_{j+1} = u_j + Le_j$$

where the systems error signal is defined as $e_j = y_d - y_j$ with j the period iteration index.

4.4 ILC Theory

The theory behind iterative learning control revolves around its ability to produce asymptotic stability in the controlled system [3]. While the SS200 system is already stabilized by its onboard feedback controller, the desire is to increase its efficiency with the addition of feedforward control. In this case, a reduction in the system response error correlates to less excess energy delivered and an increase in power efficiency. The

SS200 feedback system with the addition of a parallel type ILC structure is presented mathematically below (4.2), and its accompaning block diagram representation is shown in Figure 4.7 [3].

$$y_j(t) = \frac{G(q)}{1 + G(q)C(q)} u_j(t) + \frac{1}{1 + G(q)C(q)} d(t) \tag{4.2}$$

Here t is the time index, j is the period iteration index, q is the forward time shift operator $qx(t) \equiv (t + 1)$, y_j is the system output, u_j is the control input to the plant, $d(t)$ is an exogenous signal that repeats with each period iteration, $C(q)$ is the feedback controller, and $G(q)$ is the plant posed as a proper rational function of q with relative degree, m (assumed to be 1) [3]. The learning controller algorithm u_{j+1} with Q-filter, $Q(q)$, and learning function, $L(q)$, takes the form,

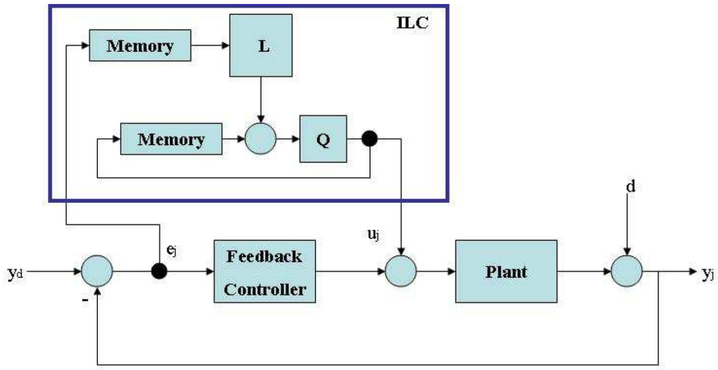

Figure 4.7: SS200 Feedback Model with Parallel ILC Addition

$$u_{j+1}(t) = Q(q)[u_j(t) + L(q)e_j(t + 1)] \tag{4.3}$$

$L(q)$ is the critcal component in (4.3) for improving the system's performance. Variations of the learning function can extend anywhere from linear functions to non-linear functions, time varying functions, H_∞ methods, and even quadratically optimal designed functions [3] [1]. For the ILC case presented in this thesis an "inverted plant" learning function has been chosen for implementation. The Q-filter can be used if desired to emphasize certain frequencies in the learning process. Here, the Q-filter is set as unity but remains incorporated throughout the theory for clarity and completeness.

The "inverted plant" of the ILC scheme differs from that previously discussed in the *ideal* feedforward example. ILC calculations only require the error of the systems response and not the actual disturbance signal itself. Consequently, a new relation equation for the "inverted plant" learning function is also necessary. Analysis of Figure 4.7

leads to an effective system relation between the measured error signal and the input of the new control signal as (in the frequency domain[1]),

$$G(z)S(z) = \frac{G(z)}{1 + G(z)C(z)} \tag{4.4}$$

where $S(z)$ is the system's *Sensitivity Function* defining the relation between the disturbance input and the error signal, and the plant transfer function $G(z)$ defining the relation between the ILC control input and the disturbance input.

$$S(z) = \frac{1}{1 + G(z)C(z)} \tag{4.5}$$

Therefore, the "inverted plant" in this case actually refers to the inverse of (4.4), or the inverse of the plant times the sensitivity function. Note that all calculations for the feedforward signal are performed in the frequency domain. The frequency domain representations of the closed loop system (4.2) and the learning algorithm (4.3) are provided below.

$$Y_j(z) = \frac{G(z)}{1 + G(z)C(z)}U_j(z) + \frac{1}{1 + G(z)C(z)}D(z) \tag{4.6}$$

$$U_{j+1}(z) = Q(z)[U_j(z) + zL(z)E_j(z)] \tag{4.7}$$

where $E_j(z) = Y_d(z) - Y_j(z)$

4.4.1 Convergence

With the control signal update (4.3), the system is stable according to [3] with converging control if there exists $\bar{u} \in \Re$ such that $|u_j(t)| \leq \bar{u}$ for all $t = \{0, \ldots, N-1\}$ and $j = \{0, 1, \ldots, \}$, and, for all $t \in \{0, \ldots, N-1\}$

$$\lim\nolimits_{j\to\infty} u_j(t) \quad \text{exists.}$$

Where here, convergence control is defined as $u_\infty(t) = lim_{j\to\infty} u_j(t)$. This optimized control signal $u_\infty(t)$ brings the system output $y_j(t)$ as close to the reference value $y_d(t)$ as possible on the given interval.

[1] The one-sided z-transform of the signal $\{x(k)\}_{k=0}^{\infty}$ is $X(z) = \sum_{k=0}^{\infty} x(k)z^{-1}$, with the z-transformation of a system being obtained by replacing q with z. The frequency response of a z-domain system is given by replacing z with $e^{i\theta}$ for $\theta \in [\pi, -\pi]$,[3]

Analysis of this stability condition performed on the frequency domain representations of the SS200, (4.6) and (4.7), can be seen below. Substituting $E_j(z) = Y_d(z) - Y_j(z)$ with (4.6) into (4.7) yields the iteration domain dynamics equation,

$$U_{j+1}(z) = Q(z)[1 - zL(z)G(z)S(z)]U_j(z) + zQ(z)L(z)[Y_d - S(z)D(z)] \tag{4.8}$$

A sufficient condition for convergence of this system is obtained with the following requirement.

Theorem 4.1 [3][11]

If

$$\| Q(z)[1 - zL(z)G(z)S(z)] \|_{\infty} < 1, \tag{4.9}$$

then the ILC system with $N = \infty$ is asymtoptically stable (AS).

With Theorem 4.1 satisfied the learning control $u_j(t)$ converges to its optimal value $u_\infty(t)$. A more generalized definition of this theorem can be found in [11]. Equation (4.9) is the convergence criterion that must be satisfied by the learning function L to guarantee that the system's response error converges to zero. Here the learning function L is re-defined as $\hat{F}^{-1}$ to represent the inverted estimation of the closed loop relation noted by (4.4), and is defined below in its frequency domain representation.

$$\hat{F}(z) = \frac{\hat{G}(z)}{1 + \hat{G}(z)C(z)} \tag{4.10}$$

If $\hat{F}(z)$ is an exact representation of $G(z)S(z)$ then the evaluation of $\hat{F}^{-1}(z)G(z)S(z)$ will be unity for all frequencies of interest, thus (4.9) will equate to zero and satisfy the convergence criterion. Such an estimate will provide for error convergence to zero within one iteration of feedforward control as guaranteed by the convergence criterion. Proof of this is demonstrated in the following section on performance and furthermore in [11]. $\hat{F}(z)$ estimations close to (4.4) that also satisfy the criterion but do not evaluate identically to zero will require further iterations before yielding full error convergence. As the learning iterations which estimate $\hat{F}(z)$ converge so will the resulting system output signal [11].

$$\lim_{j\to\infty} y_j(t) = y_d(t)$$

4.4.2 Performance

The actual effectiveness of ILC is determined from analysis of the resulting asymptotic error. This can be done qualitatively from system response observations or quantitatively in comparison of the root mean square (RMS) of the error. If the system under consideration satisfies the conditions for AS as noted in the convergence section, then the asymptotic error is [3] [11],

$$\begin{aligned} e_\infty(t) &= \lim_{j\to\infty} e_j(t) \\ &= \lim_{j\to\infty} (y_d(t) - y_j(t)) \\ &= \lim_{j\to\infty} (y_d(t) - G(q)S(q)u_j(k) - S(q)d(t)) \\ &= y_d(t) - G(q)S(q)u_\infty(t) - S(q)d(t)). \end{aligned} \tag{4.11}$$

After substituting $u_\infty(t)$ into the equation above and solving for $e_\infty(t)$ the frequency domain error function translates to,

$$E_\infty(z) = \frac{1 - Q(z)}{1 - Q(z)[1 - z\hat{F}^{-1}(z)G(z)S(z)]}[Y_d(z) - S(z)D(z)] \tag{4.12}$$

with the necessary and sufficient condition for convergence to zero error as follows.

Theorem 4.2 [3]
Suppose G, C, and F are not identically zero. Then, for the ILC system, $e_\infty(t) = 0$ for all k and for all y_d and d, if and only if the system is AS and $Q(q) = 1$.

Iteration error of the system with the addition of the parallel ILC structure is given by (4.13).

$$e_j(t) = -G(q)S(q)u_j(t) - S(q)d(t) \tag{4.13}$$

Verification of this error convergence is described by (4.14) for the 'next iteration' error that follows,

$$e_{j+1}(t) = [1 - \hat{F}^{-1}(q)G(q)S(q)]e_j(t) \tag{4.14}$$

Notice that this equation evaluates to

$$e_{j+1}(t) = e_j(t) - \hat{F}^{-1}(q)G(q)S(q)e_j(t)$$

where if

$$\hat{F}^{-1}(q)G(q)S(q) = 1$$

then we are left with

$$e_{j+1}(t) = e_j(t) - e_j(t)$$

so that the next iterations error becomes zero.

$$e_{j+1}(t) = 0$$

Any estimates of the learning function $\hat{F}$ that satisfy (4.9) over the frequency range of interest will produce a next iteration error value that will continually degrade until it has converged to zero. Notice from the evaluation following (4.14) that if $\hat{F}^{-1}(q)G(q)S(q)$ is less than unity then some error will remain in the next iteration. Conversely, if the criterion is not satisfied and the evaluation is greater than one, the error in the system will begin to grow. The successfully updated ILC control signal works in addition to the feedback control signal to further improve on the systems transient and steady state responses. In the next section, the procedure used to estimate the ILC learning function $\hat{F}$ is discussed.

4.5 Learning Function Estimation

Plant dynamics are often not completely known to the control designer. In these cases identification techniques can be used to provide an accurate approximation of what the system characteristics are. For the ILC system described in the previous section, estimation of $\hat{F}(q)$ is a critical step in building a successful learning controller. If the characteristics of the learning function are not robust enough to satisfy the convergence criterion then no error convergence will take place. In fact, if the convergence criterion is not satisfied then the addition of ILC will result in a growth of system output error towards infinity. The system model used for simulation has been setup in such a way that the structure of the "plant inversion" type learning function involves much more than just the plant. The relation between the measured error signal, disturbance signal input, and feedforward signal input calls for an "inverted plant" that is the inverse of (4.4). In order to utilize this inverted plant expression it must first be identified. This section takes a slight step away from ILC theory to discuss the method used to estimate this expression and the varying results that were obtained.

4.5.1 Linear Regression Method

In order to identify any type of system one will need some measurements of both the input signal to the system as well as the output response of the system. The identification procedure followed here begins with a measured step input signal injected into the system as pictured in Figure 4.8 along with measurements of the system's output response. The two signals are then used in an Auto-Regressive eXogeneous (ARX) system identification technique to given an estimate of the closed-loop system.

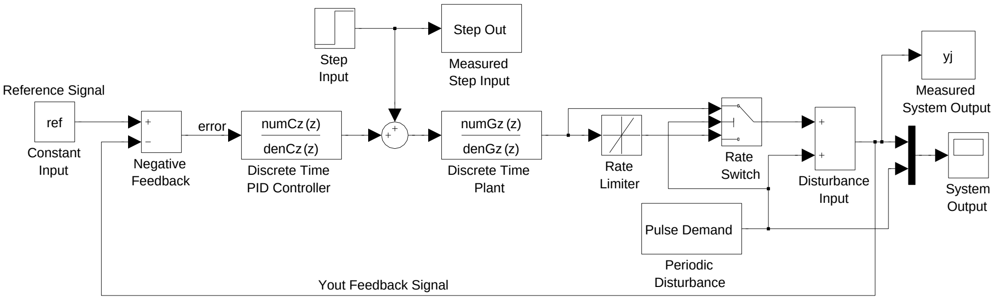

Figure 4.8: SIMULINK Model for Step Response Data Collection

The SISO relation from the measured signals is arranged in difference equation form as seen below; this is also referred to as an ARX model [10].

$$\begin{aligned} y(t) + a_1 y(t-1) + \ldots + a_{n_a} y(t-n_a) \\ = b_1 u(t-1) + \ldots + b_{n_b} u(t-n_b) + e(t) \end{aligned} \tag{4.15}$$

With θ defined as a vector of the unknown system parameter coefficients each of order n_a and n_b respectively [10] [11],

$$\theta = [a_1 \;\; a_2 \; \ldots \; a_{n_a} \;\; b_1 \; \ldots \; b_{n_b}]^T \tag{4.16}$$

$A(q)$ and $B(q)$ can be defined as

$$A(q) = 1 + a_1 q^{-1} + \ldots + a_{n_a} q^{-n_a}$$

$$B(q) = b_1 q^{-1} + \ldots + b_{n_b} q^{-n_b}$$

the transfer function form of the estimate including error becomes (4.17) and is given here along with an illustrative

$$G(q,\theta) = \frac{B(q)}{A(q)}, \qquad H(q,\theta) = \frac{1}{A(q)} \tag{4.17}$$

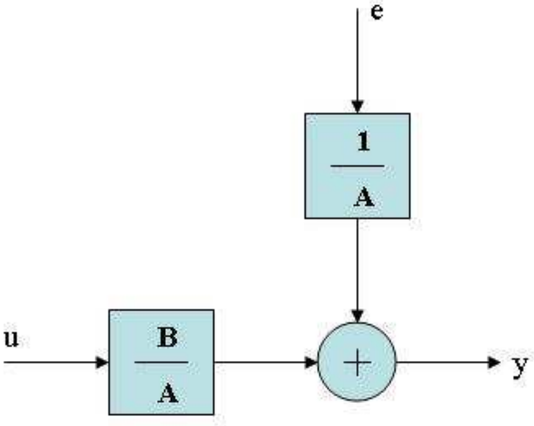

Figure 4.9: ARX Model Structure [10]

block diagram, Figure 4.9.

With the error term considered to be insignificant the input $u(t)$, ouput $y(t)$, $B(q)$, and $A(q)$ vectors can be put into the prediction equation form below

$$\hat{y}(t|\theta) = B(q)u(t) + [1 - A(q)]y(t) \tag{4.18}$$

By introducing the regression vector defined as

$$\psi(t) = [-y(t-1) \ \ldots -y(t-n_a) \ \ u(t-1) \ \ldots \ u(t-n_b)]^T \tag{4.19}$$

the prediction equation takes on a form which is linear in θ [10].

$$\hat{y}(t|\theta) = \theta^T \psi(t) = \psi^T(t)\theta \tag{4.20}$$

The prediction equation utilizes a least-squares estimate to determine the best fitting combination of parameter coefficients to match the input/output data provided and system order specified. This process is followed to determine the system estimate $\hat{F}$ which best approixmates the inverse of GS. In this fashion $\hat{F}^{-1}$ becomes the solution of the problem,

$$min \parallel 1 - \hat{F}^{-1}GS \parallel$$

which directly determines the next iterations error as given in (4.14) [11]. The system order can be varied in experiments to determine the range of functions that will accurately model the system characteristics. A brief overview of the least-squares criterion used is provided in the appendix. The reader is referred to [10] and [2] for further insight.

Fortunately in our situation one can compare the learning function estimations with the actual expression using knowledge of the plant and controller equations. The PID controller used here is represented by a 2^{nd} order function, and the plant approximation is a 6^{th} order function, thus (4.4) when evaluated becomes an 8^{th} order function. Experimental estimations of (4.4) range from as low as a 2^{nd} order function to that of an 8^{th} order for analysis purposes. The ultimate goal of this experimentation is to identify the lowest order model estimate that will satisfy the convergence criterion and still provide for a relatively quick convergence of system error. The actual 8^{th} order closed loop system in discrete form is presented here for reference.

$$GS = \frac{b_0 z^8 + b_1 z^7 + b_2 z^6 + b_3 z^5 + b_4 z^4 + b_5 z^3 + b_6 z^2 + b_7 z + b_8}{a_0 z^8 + a_1 z^7 + a_2 z^6 + a_3 z^5 + a_4 z^4 + a_5 z^3 + a_6 z^2 + a_7 z + a_8} \tag{4.21}$$

Table 4.1: Discrete Time Closed Loop ILC Relation Coefficients

Numerator	Denominator
b_0 = 15	a_0 = 1
b_1 = -116.1	a_1 = -7.713
b_2 = 393.8	a_2 = 26.05
b_3 = -763.6	a_3 = -50.34
b_4 = 926.3	a_4 = 60.87
b_5 = -719.9	a_5 = -47.17
b_6 = 350.1	a_6 = 22.87
b_7 = -97.38	a_7 = -6.347
b_8 = 11.86	a_8 = 0.7718

Response Comparisons

The magnitude response plot of $G(z)S(z)$ (dashed black line) is compared with various order estimates obtained from the ARX identification procedure in Figure 4.10. The order estimates displayed were all evaluated in (4.9) and proved to satisfy the criterion over the entire frequency range of interest, (0–100kHz). Those lower order estimates not satisfying the criterion are ommitted here. The magnitude responses of the convergent order estimates show adherence to the general form of the magnitude repsonse of $G(z)S(z)$. For reference, the individual transfer function equations of these estimates can be found in the appendix. Convergence criterion analysis is detailed in the following section.

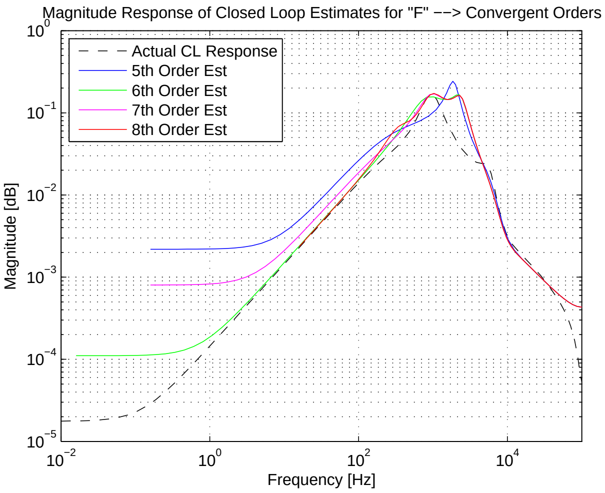

Figure 4.10: Magnitude Response Comparison of Convergent Estimates

Convergence Evaluations

It should be noted that any preliminary testing of these identification and control techniques should be performed in a simulation software package such as MATLAB. Therefore, some knowledge and accuracy of the plant and overall system dynamics will be known beforehand. In this case, the controller of the system was known but plant dynamics had to be identified. The plant model $G(z)$ used for the baseline comparison here was computed in Chapter 2 and verified as an accurate representation of the SS200 plant.

Results of the convergence criterion analysis (4.9) using the ARX estimated systems are displayed in Figure 4.11. The 5^{th}, 6^{th}, 7^{th}, and 8^{th} order systems are color coded for easy observance of their respective behaviors. Notice how each order estimate remains nicely under the threshold value of 1 over the entire frequency range considered. The results of these experiments prove that a successful closed-loop estimate for $G(z)S(z)$ is required to be at least a 5^{th} order function, while an 8^{th} order estimate is nominal. The 8^{th} order estimate is considered nominal due to prior knowledge of the order of $G(z)S(z)$, its magnitude response similarity to that of $G(z)S(z)$, and its quick drop off observed in its convergence criterion analysis.

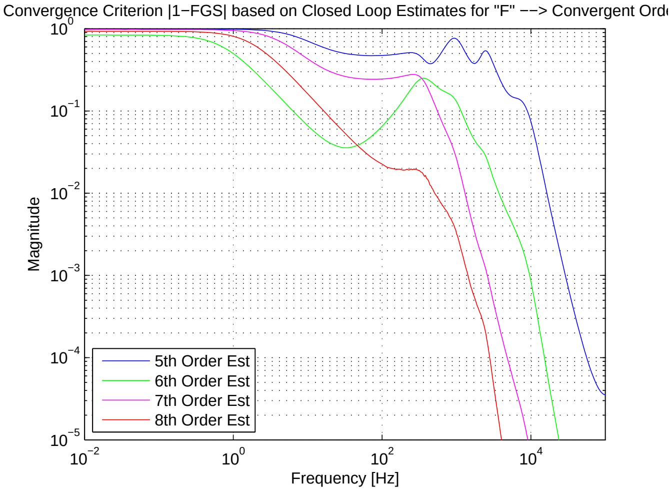

Figure 4.11: Convergence Criterion Analysis of Convergent Estimates

4.6 ILC Implementation

An ideal implementation of the ILC scheme would involve on online learning system in which the new control signal would be updated while the power supply is running. The SS200 however, does not currently have the capacity to perform online floating point calculations in the relatively short time between demand periods. Due to these limitations the learning controller is instead calculated offline and done so here for one constant demand signal chosen for demonstration.

To begin, the modeled SIMULINK system is allowed to run for ten periods and then stopped. In this particular case each period is 15ms long for a total duration of 150ms. The purpose of the ten period iteration is to allow any initial conditions in the system to die out and to obtain the cleanest measurement data possible. MATLAB round-off calculation error leads to some minor problems in fulfulling the repeated initial condition requirement of ILC theory. Performing new control calculations and implementations every tenth period in the simulation reduces interference caused by these round-off errors. The last periods error is captured and used to calculate the new feedforward control signal to be injected into the system. In the frequency domain, the error signal is run through the inverse of the estimated system dynamics $F(z)$, and the inverse fourier transform of that signal is then modified slightly before entering into the system and adding to the existing control signal. Modification of the calculated feedforward signal is done to ensure minimal interference from this signal during the disturbance

"OFF" time. Non-linear behavior in the SS200 during this time frame conflicts with the linear ILC theory that is being implemented on it. For this reason, the ILC signal is only applied to the system during the "ON" time and also during a slight time frame before that. Activation of the ILC signal slightly before the pulse "ON" time allows the feedforward nature of the signal to preemptively react to the disturbance. See Figure 4.12 for an illustrative example. This modification involves redefing the feedforward signal's values during the "OFF" time frame as being equal to its sample value at 500 samples from the end of the signal; or about half the "ON" time from the end. For the 15ms disturbance period with 5ms "ON" time and sampling frequency of 200kHz this equates to setting the feedforward samples 1001–2500 equal to the value of sample 2500. By doing this, the effects of the ILC during the "OFF" time are minimal as well as the slight disturbance caused from calculation round-off error during its re-introduction. The modeled system is run again, but this time the duration is for 20 periods with the previously calculated feedforward signal injected into the system at the beginning of the 11th period. Observation and numerical analysis of the resulting system error can be used in order to determine whether or not further ILC iterations are necessary. If so, then the 20th periods' error is used to calculate an additional feedforward signal in the same manner as before. This new signal will be added to the system at the beginning of the 21st period. It should be noted that for this particular application once these signals are injected into the system they are expected to repeat periodically until the system

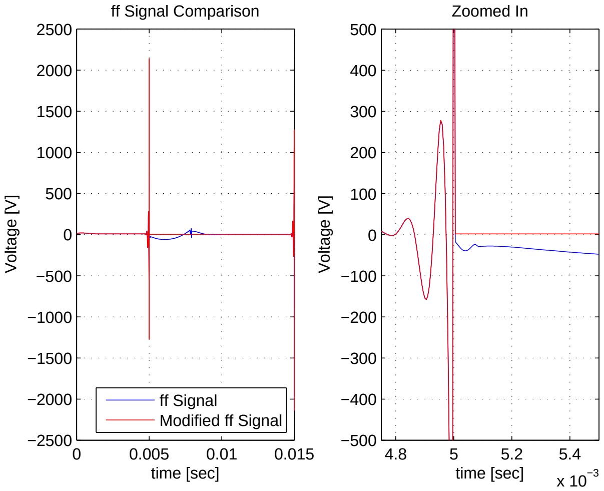

Figure 4.12: FeedForward ILC Signal vs. Modified FeedForward ILC Signal

is shut off. If further error reduction is needed, the process will continue in the same manner with the system stopped every consecutive 10^{th} period until the attenuation of error is satisfactory.

4.7 ILC Results

Immediate results of the ILC's performance varies with the magnitude order of the learning function estimate. The 2^{nd}, 3^{rd}, and 4^{th} order estimates all failed to satisfy the convergence criterion and therefore their results will not be shown here. Successful error attenuations from the higher order 8^{th} and 5^{th} learning function estimates are provided here. Each performance result plot displays the effects of the implemented ILC control for one iteration of calculations. The SIMULINK model is set up for a reference level of 5 volts, a demand signal of 2.5 amps for 5ms, and a total period length of 15ms. The criterion for comparison of the results is similar to that in the Anti-Windup section, however here the increase in power efficiency is the only major concern and not the system settling time. The regulators energy use over time is calculated with the equation:

$$P = \frac{1}{T}\int_0^T \frac{V^2}{R}\,dt$$

wherere $T = 5$ms, $dt = 5e^{-6}$s, and $R = 2$ Ohms. The baseline power rate calculated for the "ON" time duration is 13.68 watts equating to 1.18 watts of excess power provided for each demand pulse. This represents an efficiency of 91.37%, which is significantly higher than the "OFF" time performance. The system response with the addition of one iteration of the 8^{th} order ILC estimate is shown in Figure 4.13. Notice that the transient oscillations during the demand "ON" time have been virtually eliminated (green line). The calculated power rate for the "ON" time frame is 12.522 watts, which is an outstanding improvement of the power rate to a near nominal performance. With the addition of the 8^{th} order ILC the "ON" time efficiency is now 99.82%. This resulting performance demonstrates the power of the ILC control method when the learning function is estimated correctly. In just one iteration from the iterative learning controller the systems performance has been improved to nearly 100% efficiency. The first iteration systems error with the 5^{th} order estimate in (Figure 4.14) improves from the original, but its response still shows some oscillating behavior. The error remaining calculates to a power rate of 13.13 watts which is an improvement from the baseline yielding an

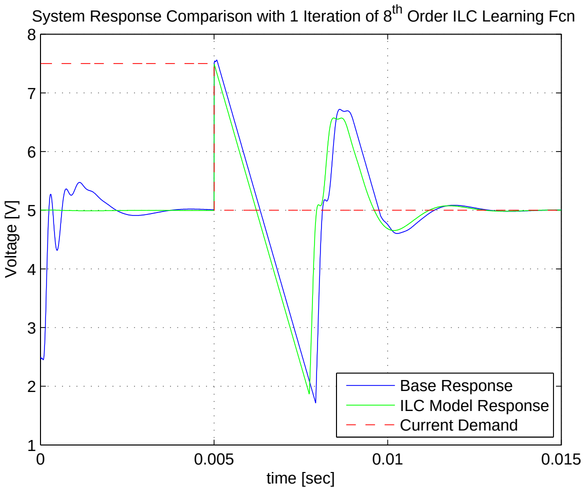

Figure 4.13: System Response from additional 8^{th} Order Estimated Learning Function

increase in efficiency to 95.22%. This lower improvement in performance was expected due to the lower order used in estimating the learning function. Further iterations not shown here result in the 5^{th} order estimate improving system performance to near 100% efficiency after four iterations.

The reduction in system error during the “ON” portion of the pulse disturbance demonstrates the theoretical effectiveness of the ILC. For the case of this SS200 system model, an estimated system of at least the 5^{th} order is needed to satisfy the convergence criterion. Results of this section clearly show that the system error during the demand pulse can be eliminated completely and that an 8^{th} order ILC estimate can increase the “ON” time efficiency up to 99.82%.

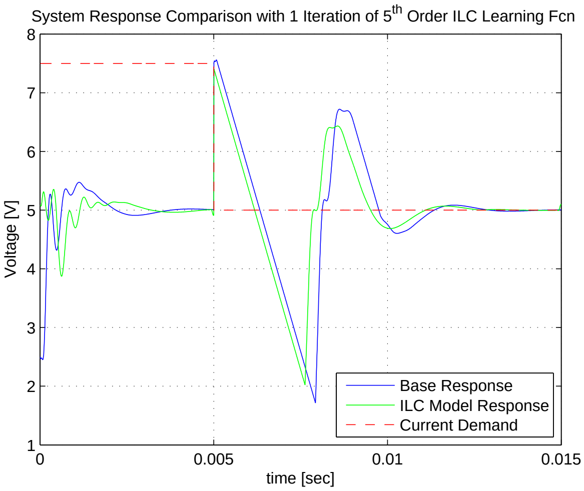

Figure 4.14: System Response from additional 5^{th} Order Estimated Learning Function

Chapter 5

Combined Control Solution

5.1 Enhanced Control Package

Due to the switched linear and non-linear effects that the power supply exhibits the two problem areas addressed in this thesis were done so in a separate fashion. This approach resulted in two different solutions which were also implemented seperately. The goal of this chapter is to convey how the two methods tested can be applied in such a way that each can function simultaneously with one another to enhance the overall systems performance. The SIMULINK model is modified to accommodate both ILC and Anti-Windup additions as shown below. The ILC portion is highlighted in blue while the Anti-Windup portion is highlighted in red. The following sections lay out the

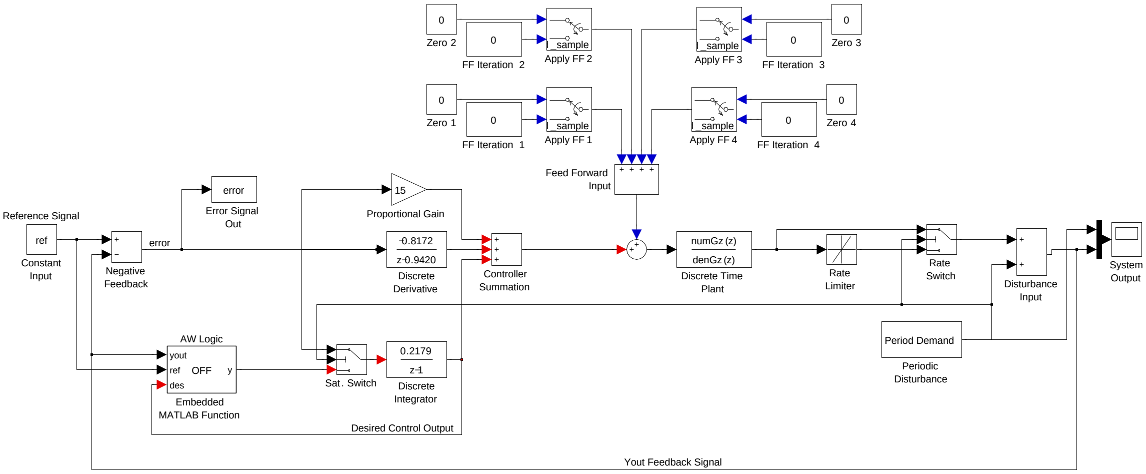

Figure 5.1: SIMULINK Model with ILC and Anti-Windup Modifications

procedures for applying ILC and Anti-Windup logic together along with the results of their combined application.

5.1.1 Procedure

Initially the SIMULINK model should be set up to run for a length of time representing 10 period iterations of the disturbance signal. The Anti-Windup logic should be turned off and no additional feedforward signals should be acting on the system during this time. With the system's parameters loaded into MATLAB one can then follow the ILC procedure as described in section 4.6. The ILC iteration is followed until the updated feedforward signals which satisfy the users' criterion for "ON" time error attenuation have been calculated and stored. The system will then need to be stopped again so that the calculated feedforward signals can be setup for their appropriate delayed injection time into the system. Finally, also at this time, the Anti-Windup code can be activated to filter the integrators error feed signal with a generous initial bound limit.

Following the initial limits of the Anti-Windup logic from Chapter 3, a set value of $U_{min} = -20$ volts is used first with the 8^{th} order estimation for $\hat{F}$. In such cases as a lower order estimation is used other than that of the actual system order, one is cautioned to first initialize the Anti-Windup limits to a slightly higher value. The system is then run for a duration of at least 10 more periods beyond the instance where the last feedforward signal injection occured. Observing the response during this time will allow the user to tune the limits of the Anti-Windup logic to a value yielding the best results. An overly tight bound limit may lead to adverse reactions when applied with the ILC feedforward signal. In experimentation and testing, such a tight constraint prevented the system from ever fully returning to the reference value desired. Initializing the Anti-Windup limits to a larger value and then progressively reducing that value based upon observance of the systems performance is a simple and effective task.

5.2 Combined Results

In order to further demonstrate the effectiveness of combining Anti-Windup logic with iterative learning control an example trial is worked through next. This example should help to clarify the procedure required to combine and apply the two methods on the simulated system. The test case makes use of the nominal 8^{th} order system estimate as previously determined.

Following the identification procedure in section 4.5 an 8^{th} order system estimate $\hat{F}$ with a single delay is given by

$$\hat{F}(z) = \frac{b_0z^7 + b_1z^6 + b_2z^5 + b_3z^4 + b_4z^3 + b_5z^2 + b_6z^1 + b_7}{a_0z^8 + a_1z^7 + a_2z^6 + a_3z^5 + a_4z^4 + a_5z^3 + a_6z^2 + a_7z + a_8} \tag{5.1}$$

The SIMULINK model in Figure 5.1 is run 150ms or 10 periods of the disturbance signal.

Table 5.1: Discrete Time Estimated Closed Loop Relation Coefficients

Numerator	Denominator
b_0 = -0.0006563	a_0 = 1
b_1 = 0.004751	a_1 = -7.713
b_2 = -0.01479	a_2 = 26.05
b_3 = -0.02676	a_3 = -50.34
b_4 = 0.01679	a_4 = 60.87
b_5 = -0.005866	a_5 = -47.17
b_6 = 350.1	a_6 = 22.87
b_7 = 0.0008799	a_7 = -6.347
b_8 = 0	a_8 = 0.7718

The last period's error signal is collected and displayed below. Using this error signal and the $\hat{F}$ estimate above the first iteration modified learning control signal is calculated, Figure 5.3. The control signal is tested in the modified SIMULINK model Figure 5.1 for a total of 20 periods (300ms). Observation of these results, Figure 4.13, show satisfactory attenuation of "ON" time error such that further testing with the addition of Anti-Windup logic can begin. The system is now restarted with the feedforward signal set to inject at the appropriate time, and the Anti-Windup logic activated with the U_{min} limit set at −20 volts. Results of this combined run are seen in Figure 5.4, and have been

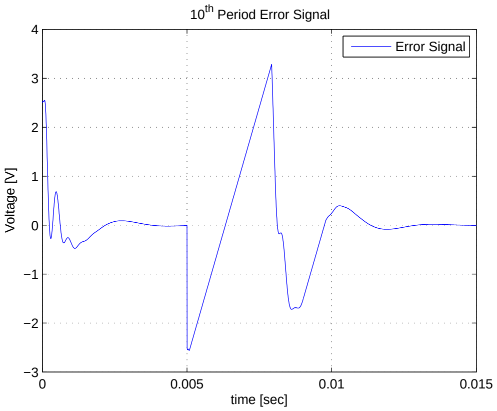

Figure 5.2: System Error for 8^{th} Order Estimate

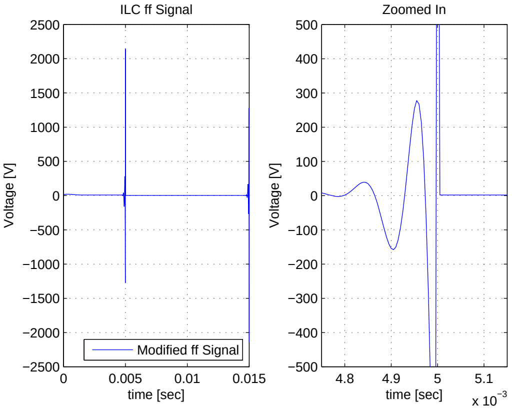

Figure 5.3: 8^{th} Order Estimate 1^{st} Iteration Control Signal

zoomed in to show the last period of the combined 20 period test duration. It is clear from the green line in Figure 5.4 that both the iterative learning control signal and Anti-Winup logic are functioning together successfully to attenuate the systems error signal. The iterative learning controller has eliminated the error during the “ON” time as seen in chapter 4, and the “OFF” portion has been improved upon by the Anti-Windup control as seen in chapter 3. With the combined improvements from both additional control schemes the new power rate calculated for the overall system response is 14.02 watts. The efficiency of the system over the entire demand period is now 89.15% as opposed to the efficiency of the baseline model which is 80.4%. 2% settling time has also been reduced to 5.3 ms from the original 6.1 ms. Some amount of undershoot still occurs in the output response but has been fairly attenuated. Further improvement on the system

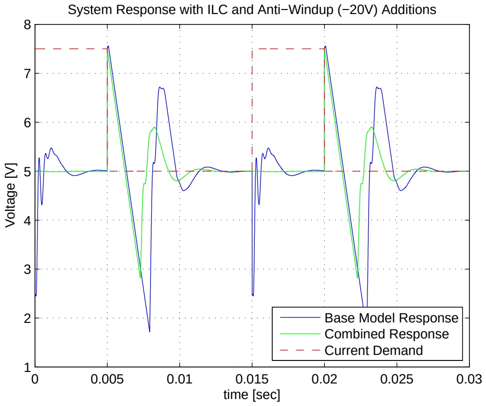

Figure 5.4: 8^{th} Order $\hat{F}$, 1^{st} Iteration plus AW Bound −20 V Combined Results

can be obtained with a stricter bound limit. Next, the U_{min} limit will be set at -1 V to yield a better output performance with the combined control system. Figure 5.5

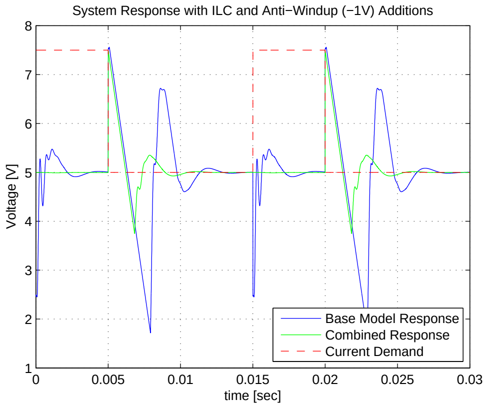

Figure 5.5: 8^{th} Order $\hat{F}$, 1^{st} Iteration plus AW Bound -1 V Combined Results

is clearly a better improvement over Figure 5.4 with the undershoot of the response reduced even further. Here, the settling time shrinks to 3.64ms, and the power rate of the new response is calculated as 13.45 watts. This represents an efficiency of 92.9%, which is a 12.5% overall increase in the efficiency of the system. A great improvement by any standards.

The combined addition of ILC and Anti-Windup logic to the modeled system successfully improves the areas of performance that were targeted. While some slight modifications to the ILC control signal were required it has been shown that the two methods can function in a combined effort. With the ILC eliminating the “ON” time error and the Anti-Windup logic attenuating the “OFF” time error the efficiency of the modeled system is increased by 12.5%. This increase puts the model in the low 90% range of efficiency which is significantly better than *most* modern DC-DC regulators.

Chapter 6

Simplification of Ideal Feedforward via Pulse Signals

6.1 Introduction

Unfortunately, the mathematical sophistication of the ILC technique requires a greater amount of computing power than is readily available on the current SS200 system. Therefore, in order to further motivate the development and implementation of the demonstrated ILC technique an experimental comparison test with a simpler design is performed and analyzed. This test compares the benefits of using the more robust ILC scheme versus a simpler feedforward pulse design. The choice of a structured pulse signal is due to the ease in which it can be applied to the current SS200 system and its similar characteristics to the disturbance and resulting ILC signals. This chapter gives an overview of the test experiment performed, explanation of the feedforward pulse structure, and brief discussion of the concluding results. A criterion is also established in order to weigh the two methods against each other and give insight to the potential increase in efficiency that each is capable of providing.

6.2 Test Setup

All experimental testing is performed on the same SS200 model as described in Chapter 2 with the addition of a feedforward input block between the controller and plant, see Figure 6.1. The input block allows for either the ILC or structured pulse feedforward signal to be injected into the system. The time frame of interest for these

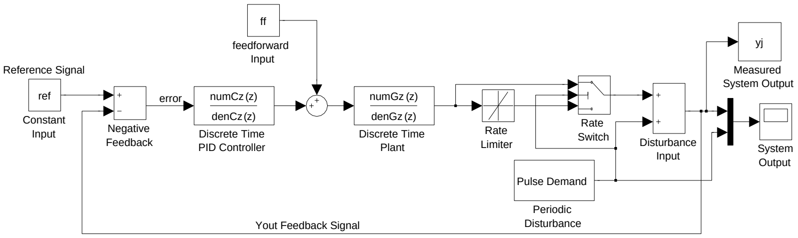

Figure 6.1: Block Diagram of Comparison Test Setup

tests is narrowed down to include only the range in which the disturbance demand signal is “ON”. Recall that this time frame is the same as that in which the ILC signal has been programmed to act on the system. The length of the pulse disturbance period along with the demand “ON” time remains the same as previously set, 15ms and 5ms respectively. The sampling time for data recording is kept at 200kHz.

The feedforward pulse structure has three main areas of interest; feedforward time, extended time, and voltage magnitude. The feedforward time represents the instance in time before the disturbance signal begins when the pulse signal is injected into the system. This same amount of time is used before the low side of the disturbance signal begins to apply the feedforward signal to the system in the opposite direction. The addition of this energy into the system is meant to counteract the sudden increase in demand that is placed on the system. The extended time refers to the extension of the first feedforward pulse into the “ON” demand time frame. Note that the feedforward and extended times are limited in their possible combinations by the duration of the “ON” time. The third and final area is the voltage magnitude of the feedforward pulse. This value remains the same in both the positive and negative directions but is varied from test to test. The various combinations of these three factors are tested to determine how well this signal can compensate for the periodic demand disturbance. An illustrative diagram of this description is provided in Figure 6.2 to help clarify the structure of the feedforward pulse signal. The first criterion used to compare and determine the strengths of the two solution methods consist of analyzing the excess energy provided by the system response over the demand “ON” time duration. The second is a measurement of the system's deviation from the reference level at the onset of the disturbance demand. The excess power is determined in the same fashion as in Chapter 4, mainly:

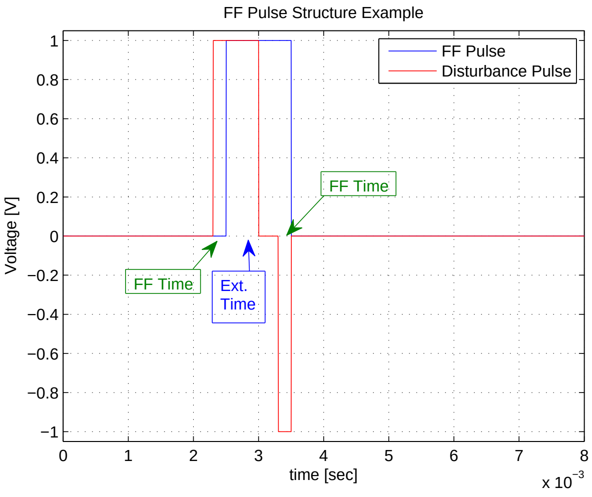

Figure 6.2: FeedForward Pulse structure compared to Disturbance

$$P = \frac{1}{T}\int_0^T \frac{(ref - V)^2}{R}dt$$

with $T = 5$ms, $dt = 5e^{-6}$s, and $R = 2$ Ohms. The system's deviation or "initial dip value" is calculated by subtracting the response voltage at the onset of the demand disturbance from the reference value. Note also that the average voltage level is documented for convenience but is not a major factor in the comparison.

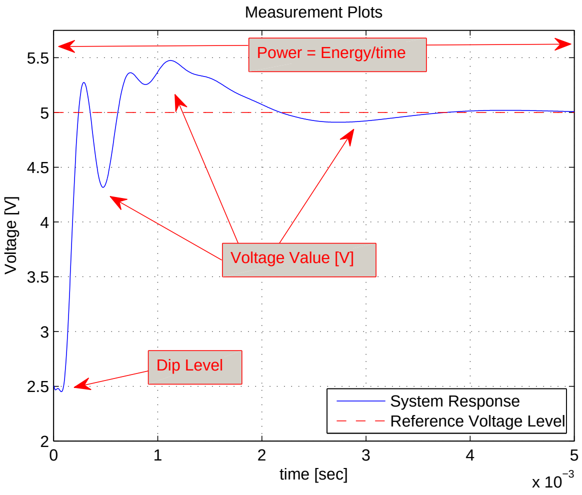

Figure 6.3: Response Measurements

6.3 Baseline and Results

The desired values in these experiments are given in Table 6.1 and represent a constant 5 volt reference level with no deviation caused from the periodic pulse demands. While the nominal values in Table 6.1 are desired, it is the baseline values exhibited by the current system with only PID control that the test results are compared against. This comparsion demonstrates the potential improvement to the system's performance that can be made with the addition of a feedforward control scheme. These baseline values have been recorded and are given in Table 6.2 along with a depiction of the systems response. Notice that the baseline dip value produced is fairly distinctive and that a return to the reference value of 5 volts is never really settled upon in the "ON" time duration.

Table 6.1: Nominal System Measurements

Power Rate:	12.5 watts
Dip Value:	0 V
Vavg:	5 V
Vmax:	5 V

Table 6.2: Baseline System Measurements

Power Rate:	13.6789 watts
Dip Value:	2.5496 V
Vavg:	4.9647 V
Vmax:	5.4740 V

6.3.1 FeedForward Pulse

The feedforward pulse experiments are performed with variations in feedforward time, extended time, and voltage magnitude value. Analysis of the complete range of these tests proves that a certain combination of times and magnitude perform better than others. The most favorable of these experimental results are consistently observed with a feedforward time of 0.2ms and an extended time of 0.8ms. The biggest reduction in the excess power rate comes with this time configuration and a pulse voltage magnitude of 9 volts. Experimental results show that increasing the magnitude of the feedforward pulse beyond this point does not lead to further improved system performance.

The resulting power rate from the optimal feedforward combination is calcu-

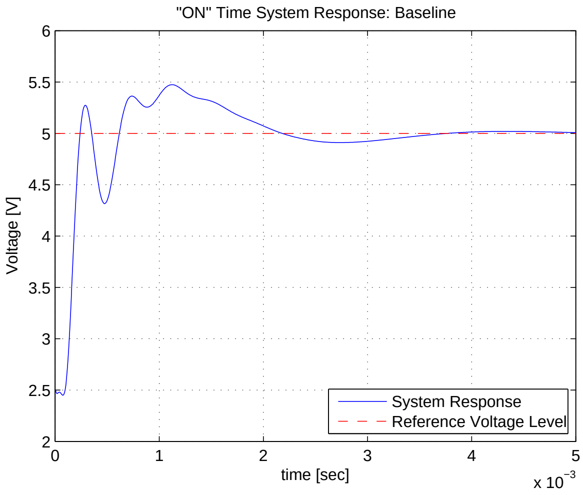

Figure 6.4: Baseline System Response

lated at 13.3389 watts which represents an efficiency of 93.71% during the “ON” time response. This demonstrates that at most, a simple feedforward pulse signal can increase this portion of the system’s response efficiency by 2.33%. Additionally, the initial dip value observed in the response has been reduced by approximately 0.654 volts.

6.3.2 ILC

Iterative Learning Control for one iteration is also implemented on the experimental setup for a direct comparison. The procedure for ILC is followed as previously documented with the results of its test run given in Table 6.4.

In Figure 6.7 one can see how the initial dip value of 0.0095 volts in has been virtually eliminated and that the response is closely held to the reference value throughout the demand time. The measurements in Table 6.4 clearly demonstrate the ILC’s effectiveness in this application. The power rate in the system is near the nominal value of 12.5221 watts, which translates to an efficiency of 99.82%. This is an 8.45% increase over the baseline value and an improvement to almost 100% efficiency during the “ON” time duration. The average voltage value held is 4.9959 volts with a maximum value of no more than 5.0027 volts. The ILC FeedForward signal overcomes the oscillations that the simple feedforward pulse still produces by providing and canceling voltage levels where need be. It is the intelligent computations of the ILC system that ensure a minimal and practically non-existent deviation from the desired reference value.

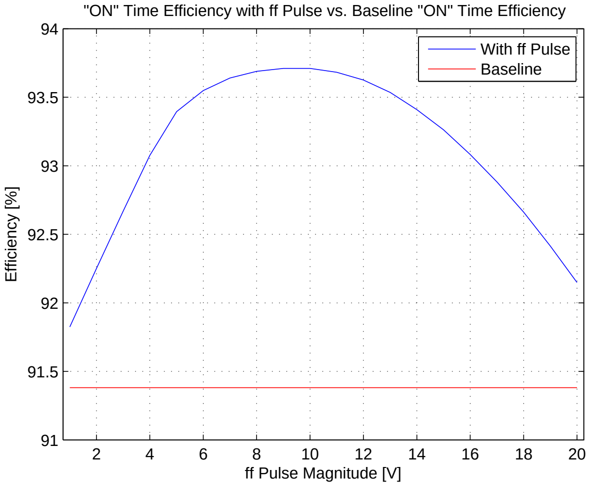

Figure 6.5: Peak FeedForward Pulse Magnitude

6.4 Test Conclusions

Both additional control schemes yielded positive results as far as their ability to increase the efficiency of the SS200 modeled system. The simpler pulse feedfoward signal displayed a surprisingly impressive reduction in system's excess power output which is promising, but still not the most effective solution. The ILC solution on the other hand can practically guarantee a close to 100% power efficiency without the numerous combination trials required of the less robust solution. Both schemes must still be implemented and tested on the actual SS200 system itself before the true results can be revealed. Fortunately the feedforward pulse structure is simple enough that it can be implemented immediately and may prove to be a good short term solution to increasing efficiency. This experiment however clearly demonstrates that that ILC solution is by far the superior method and will potentially lead to the maximum increase in efficiency possible.

6.5 Practical Considerations

Despite the outstanding performance of the ILC feedfoward signal in our application there does exist one drawback. Figure 6.8 depicts the calculated feedfoward signal required to compensate for the periodic disturbance on our system. Notice that the voltage magnitude extends from 2,100 volts to −2,100 volts. This voltage level

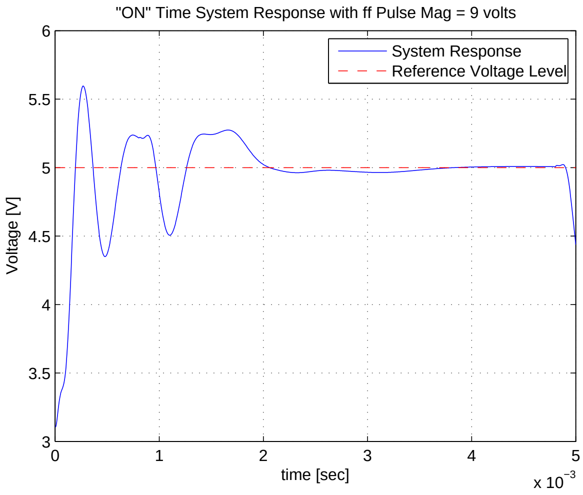

Figure 6.6: Optimal FeedForward Pulse System Response

is quite large and one not acheivable with the SS200. The large magnitude peaks of this signal are due to the pulse disturbances' sharp and near instantaneous transitions, which makes the signal difficult to compensate for. In order to reduce the magnitude of the calculated feedfoward signal one can shape the pulse disturbance to incorporate a smoother transition at its edges. Here, the edges refer to the transitioning period from low to high and vice versa. This technique is similar to that of input shaping except for in this case it is the disturbance signal which is being modified and not the actual reference signal. Figure 6.9 displays the newly calculated feedfoward signal which exhibits a range of voltage requiring a range of only -340 volts 200 volts. While these are still large voltage values, they are significantly lower than the previous value range and thus more managable.

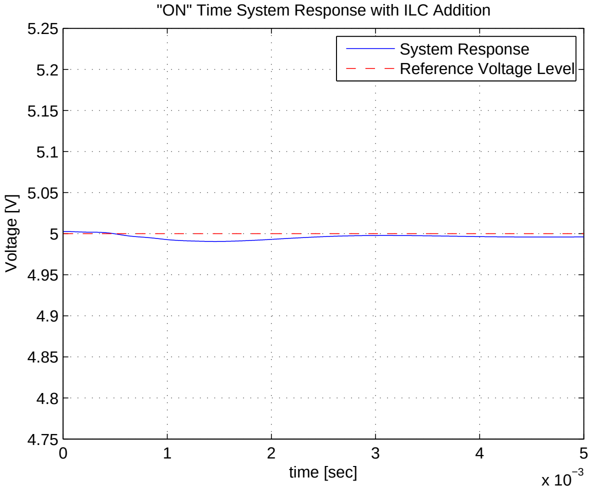

Figure 6.7: ILC System Response

Table 6.3: Optimal FF Pulse Measurements

Power Rate:	13.3389 watts
Dip Value:	1.8956 V
Vavg:	4.9544 V
Vmax:	5.5960 V

Table 6.4: ILC Measurements

Power Rate:	12.5221 watts
Dip Value:	0.0095 V
Vavg:	4.9959 V
Vmax:	5.0027 V

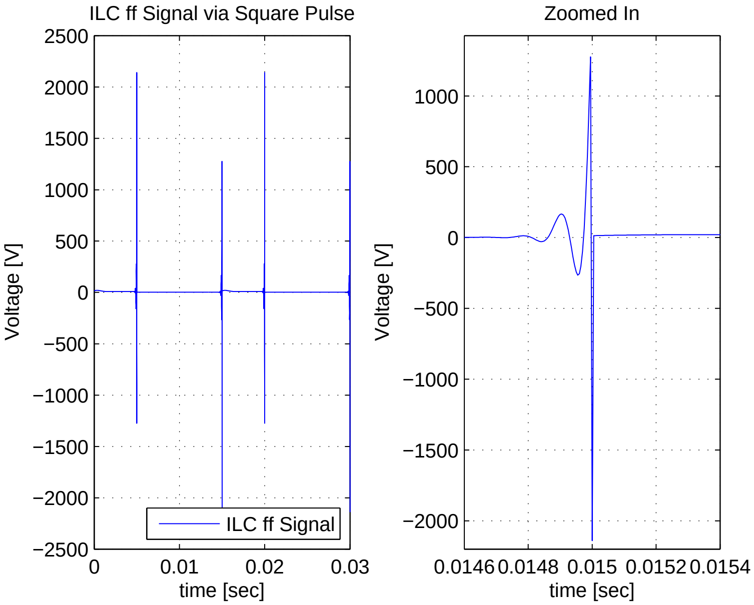

Figure 6.8: ILC Feedforward Signal via Normal Pulse

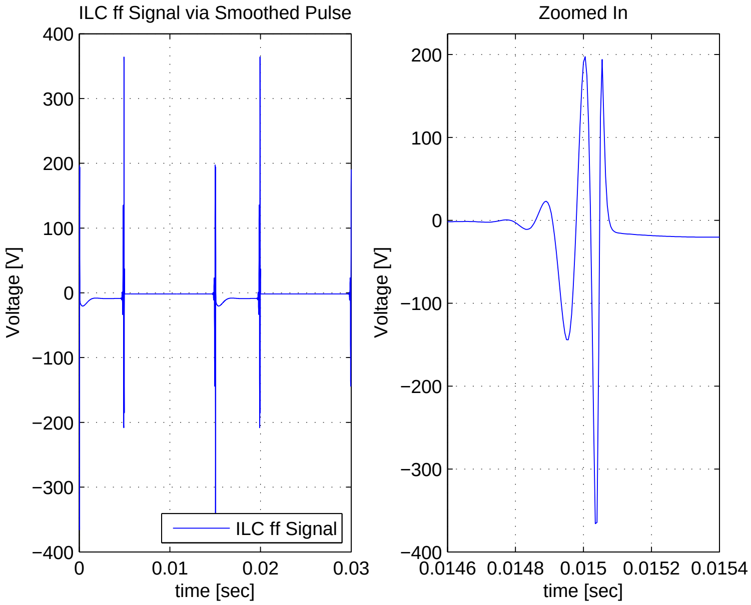

Figure 6.9: ILC Feedforward Signal via Smoothed Pulse

Chapter 7

Conclusion

A valid model representation of a real world switched power regultor is produced using advanced identification techniques and actual system response data. This model is uitilized to test combinations of further sophisticated control schemes on the system and to analyze the effectiveness of possible improvements. Two particular areas of the given power supply's performance are used as the basis for implementing enhanced control techniques, these are demand "ON" and "OFF" times. Both control schemes independently succeeded in accomplishing the tasks outlined for them. Oscillations occuring during the power supply's "ON" time frame were virtually eliminated by the applied iterative learning control technique, which confirmed the theoretical results that were anticipated. The addition of the ILC increased the model's "ON" time efficiency 99.82%. The "OFF" time improvements sought after with the Anti-Windup scheme were also demonstrated with a 3.6ms reduction in settling time, and an increase in "OFF" time efficiency from 75.81% to 88.38%.

The main results of this thesis show that it is possible to model an actual power regulator in a system such as MATLAB, and that the tandem addition of both Anti-Windup logic and Iterative Learning Control can provide improvements in that models performance. Their combined implementation is successfully observed on the simulated model derived in this thesis. Overall regulator performance is improved upon as described above with the enhanced control methodologies added into the system. The overall increase in efficiency of the modeled system increases from 80.4% to 92.9%.

7.1 Future Work

The use of advanced algorithms and controllers is soon to be the wave of the future as energy stores and availability quickly become a global concern. Theoretical results from these simulated tests have laid the ground work for implementation and real world application. The true effectiveness of these techniques can then be analyzed and hopefully implemented on various other similar types of systems. Futher advancement for this work includes generating a simulation that will enable online calculations for the iterative learning conroller as well as dynamically updating the bounds imposed by the Anti-Windup logic. The dual control scheme would then need to be adapted onto an actual power supply capable of performing the necessary calculations required for field test.

Appendix A

Estimating Coefficients

A.1 Least-Squares Criterion

The prediction error from the prediction equation in Chapter 4 is given by

$$\varepsilon(t, \theta) = y(t) - \psi^T(t)\theta \tag{A.1}$$

yielding the least-squares criterion to be minimized as

$$V_N(\theta, Z^N) = \frac{1}{N}\sum_{t=1}^{N}\frac{1}{2}[y(t) - \psi^T(t)\theta]^2 \tag{A.2}$$

The quadratic criterion above along with its linear parameterization allows for analytical minimization of θ. The least-squares estimate (LSE) for θ follows below; note that the order number which specifies the length of θ and ψ is a variable provided by the user.

$$\hat{\theta}_N^{LS} = arg\ min\ V_N(\theta, Z^N) = \frac{1}{N}\sum_{t=1}^{N}[\psi(t)\psi^T(t)]^{-1}\frac{1}{N}\sum_{t=1}^{N}\psi(t)y(t) \tag{A.3}$$

Minimized return of the θ vector produces the best fit difference equation representing the relation between the input and output signals provided.

Appendix B

Learning Function Estimations

B.1 Convergent Orders

8^{th} Order Estimation

$$\hat{F}(z) = \frac{b_0 z^7 + b_1 z^6 + b_2 z^5 + b_3 z^4 + b_4 z^3 + b_5 z^2 + b_6 z^1 + b_7}{a_0 z^8 + a_1 z^7 + a_2 z^6 + a_3 z^5 + a_4 z^4 + a_5 z^3 + a_6 z^2 + a_7 z + a_8} \tag{B.1}$$

b_0 = -0.0006563	a_0 = 1
b_1 = 0.004751	a_1 = -7.713
b_2 = -0.01479	a_2 = 26.05
b_3 = -0.02676	a_3 = -50.34
b_4 = 0.01679	a_4 = 60.87
b_5 = -0.005866	a_5 = -47.17
b_6 = 350.1	a_6 = 22.87
b_7 = 0.0008799	a_7 = -6.347
b_8 = 0	a_8 = 0.7718

7^{th} Order Estimation

$$\hat{F}(z) = \frac{b_0 z^6 + b_1 z^5 + b_2 z^4 + b_3 z^3 + b_4 z^2 + b_5 z + b_6}{a_0 z^7 + a_1 z^6 + a_2 z^5 + a_3 z^4 + a_4 z^3 + a_5 z^2 + a_6 z + a_7} \tag{B.2}$$

b_0 = -0.0006563	a_0 = 1
b_1 = 0.004095	a_1 = -6.713
b_2 = -0.01069	a_2 = 19.34
b_3 = 0.01495	a_3 = -31
b_4 = -0.01181	a_4 = 29.87
b_5 = 0.004986	a_5 = -17.3
b_6 = -0.0008799	a_6 = 5.576
b_7 = 0	a_7 = -0.7718

6^{th} Order Estimation

$$\hat{F}(z) = \frac{b_0 z^5 + b_1 z^4 + b_2 z^3 + b_3 z^2 + b_4 z + b_5}{a_0 z^6 + a_1 z^5 + a_2 z^4 + a_3 z^3 + a_4 z^2 + a_5 z^1 + z + a_6} \tag{B.3}$$

b_0 = -0.0006563	a_0 = 1
b_1 = 0.003442	a_1 = -5.718
b_2 = -0.007269	a_2 = 13.65
b_3 = 0.007721	a_3 = -17.42
b_4 = -0.004122	a_4 = 12.54
b_5 = 0.000884	a_5 = -4.823
b_6 = 0	a_6 = 0.7754

5^{th} Order Estimation

$$\hat{F}(z) = \frac{b_0 z^4 + b_1 z^3 + b_2 z^2 + b_3 z + b_4}{a_0 z^5 + a_1 z^4 + a_2 z^3 + a_3 z^2 + a_4 z^1 + a_5} \tag{B.4}$$

b_0 = -0.0006563	a_0 = 1
b_1 = 0.002834	a_1 = -4.791
b_2 = -0.004642	a_2 = 9.211
b_3 = 0.003418	a_3 = -8.883
b_4 = -0.000953	a_4 = 4.299
b_5 = 0	a_5 = -0.8354

Appendix C

Input Shaping

C.1 Modified Half Gudermann Function

The Half Gudermann Function is obtained with the equation below:

$$y = \frac{2}{\pi} \arctan(\exp^{[a(x-off)]}) \tag{C.1}$$

where the variables $a = 12$, $off = 0.6$, and x is set at a range:

$$x = \begin{cases} 0 : \pi/50 : 2\pi; , & from\ 1\ to\ 101, \\ 2\pi : -\pi/50 : 0; , & from\ 102\ to\ 202, \end{cases} \tag{C.2}$$

Some further defining of the range of x is required to produce the appropriate pulse disturbance length. This configuration however does provide for a smooth transition from “OFF” to “ON” and vice versa in approximately 0.51ms. The transition from “OFF” to “ON” is given as an example in Figure C.1 to illustrate the now smoothed edges of the pulse disturbance profile. The reader is referred to [12] for details of the Modified Half Gudermann Function.

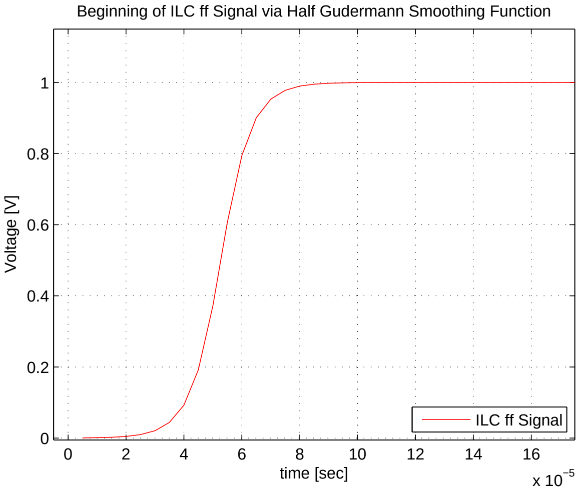

Figure C.1: Close Up of Half Gudermann Smoothing

Bibliography

[1] H.-S. Ahn, K. L. Moore, and Y. Chen. *Iterative Learning Control*, pages 4–10. Springer-Verlag London Limited, London, 2007.

[2] T. R. Bewley. *Numerical Renaissance, Simulation Optimization and Control*, pages 123–125. La Jolla, California, 2006.

[3] D. A. Bristow, M. Tharayil, and A. G. Alleyne. A Survey of Iterative Learning Control. *Institute of Electrical and Electronics Engineers Control Systems Magazine,*.

[4] S. Choudhury. Designing a TMS320F280x Based Digitally Controlled DC-DC Switching Power Supply. SPRAAB3 - Texas Instruments, 2005. Application Report.

[5] R. A. de Callafon. Estimating Parameters in a Lumped Parameter System with First Principle Modeling and Dynamic Experiments, 2003.

[6] R. A. de Callafon. Realization Algorithms, 2007. Class Resource: MAE 283A, University of California, San Diego.

[7] R. A. de Callafon, D. Roover, de Roover, and P. Van Den Hof. Multivariable Least Squares Frequency Domain Identification using Polynominal Matrix Fraction Description, 1996. In Proceedings of the 35th IEEE Conference on Decision and Control, Kobe, Japan.

[8] J. Electronics. Jaycar Electronics Reference Data Sheet. Jaycar Electronics Website, 2001. PDF Reference.

[9] G. F. Franklin, D. Powell, and A. Emami-Naeini. *Feedback Control of Dynamic Systems*, pages 610–613. Prentice-Hall, Inc., Upper Saddle River, New Jersey, 2006.

[10] L. Ljung. *System Identification*, pages 79–83. Prentice-Hall, Inc., Upper Saddle River, New Jersey, 1999.

[11] K. L. Moore. *Iterative Learning Control for Deterministic Systems*, pages 23–44. Springer-Verlag London Limited, New York, 1993.

[12] H. Morehouse. Modified Half Gudermannian Device. Beige Bag Software, Inc Website, 2008. http://www.beigebag.com/case_gud_half.htm.

[13] A. Packard et al. Dynamic Systems and Feedback, 2005. Class Notes (Ch.15): ME 132, University of California, Berkeley.

[14] A. I. Pressman. *Switching Power Supply Design*, pages 427–444. McGraw-Hall, Inc., San Fransisco, California, 1991.

[15] G. Rizzoni. *Principles and Applications of Electrical Engineering*, page 797. McGraw-Hill Companies, Inc., San Fransisco, California, 2000.

[16] M. Tham. Discretized PID Controllers. University of NewCastle Website, 1998. Study Notes on Digital Control.

[17] A. Visioli. *Practical PID Control*, pages 35–42. Springer-Verlag London Limited, London, 2006.