DYNAMIC COMPENSATION OF DIGITAL TO ANALOG CONVERTERS

By
Kirk Riley

B.S. University of Maine, 1996

A THESIS
Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Science
(in Electrical Engineering)

The Graduate School
University of Maine
August, 1998

Advisory Committee:

Donald M. Hummels, Associate Professor of Electrical and Computer Engineering, Advisor

Fred H. Irons, Castle Professor of Electrical and Computer Engineering

Ann Rundell, Visiting Professor of Electrical and Computer Engineering.

LIBRARY RIGHTS STATEMENT

DYNAMIC COMPENSATION OF DIGITAL TO ANALOG CONVERTERS

By Kirk Riley

Thesis Advisor: Dr. Donald M. Hummels

An Abstract of the Thesis Presented

in Partial Fulfillment of the Requirements for the

Degree of Master of Science

(in Electrical Engineering)

August, 1998

High accuracy Digital to Analog Converters (DACs) are becoming increasingly important as the use of digital signal processing in communications becomes more common. DACs translate a digital sequence into an analog current or voltage output. This translation is not ideal and the DAC introduces some harmonic error at its output. This thesis develops a model for DAC harmonic error, explores a low-pass measurement procedure for DAC errors, and develops a compensation scheme for compensation of these errors. Simulated results reveal the system is capable of compensating dynamic DAC errors that produce energy in a low-frequency band used for calibration. Hardware results show that harmonic distortion is reduced using this compensation scheme. However, some dynamic errors present in the DAC produce little energy in the calibration band and are not compensated.

ACKNOWLEDGMENTS

This work has been supported in part by the ARPA Digital Receiver program under a contract administered by the Office of Naval Research Grant N66001-97-C-8617.

I would like to thank my parents, William and Michelle Riley, for their unflagging emotional and financial support throughout my college career. Thank you Don Hummels for your constant guidance, assistance and operating system suggestions. Thank you Fred Irons for inspiring me to enter graduate school and opening the window of opportunity to allow me to do so as well as offering your hardware expertise. Thank you Ann Rundell for diving into unfamiliar waters and providing significant contributions to this project. Thanks to Ryan Jacobson for all his help with the hardware. Finally I would like to thank Debra Ward for her endless patience, humor, and emotional support throughout my graduate career.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Introduction

High accuracy Digital to Analog Converters (DACs) are becoming increasingly important as the use of digital signal processing in communications becomes more common. DACs translate a digital sequence into an analog current or voltage output. This translation is not ideal and the DAC introduces some harmonic distortion at its output. This thesis develops a model for DAC error, explores lowpass measurement techniques for estimating DAC error, and describes a compensation technique for removing harmonic errors for given DAC errors by adjusting the DAC input sequence.

1.1 Background

Modeling and compensating Analog to Digital Converter (ADC) errors for use in digital communication systems has been well explored [3, 4, 5, 6]. With highly linear calibrated ADCs available in digital communications systems it becomes important to remove the non-linear distortion created in these systems by DAC errors. Additionally, high linearity DACs may be used to create on-line calibration systems for ADCs. High performance DACs may also be used to build high accuracy ADCs with DACs in a feedback loop (e.g. sigma-delta ADCs). Although compensating dynamic errors at the output by adjusting input codes is a new concept, some of the techniques used to compensate ADCs can be modified for use on DACs. The frequency domain matching technique using basis functions to approximate the converter errors described in this thesis has an analogous implementation for ADCs [3]. It is shown that ADC compensation techniques can be used to form the foundation for DAC compensation techniques as presented in this thesis.

1.2 Thesis Organization

This thesis is organized into chapters that describe DAC calibration and compensation. Chapter 2 describes a DAC model with dynamic errors. It provides some background notation and justification for a dynamic error model. Finally, Chapter 2 describes how a DAC output may be compensated when DAC errors are known.

Chapter 3 describes how to estimate errors based on the model developed in Chapter 2. Chapter 3 begins with a discussion of a lowpass calibration system. It then describes how to use basis function techniques to estimate frequency domain error terms. Finally, the proper selection of DAC input calibration signals is discussed in Chapter 3.

Chapter 4 presents simulation results. These results are found by applying Chapter 3 error estimation techniques and the compensation technique of Chapter 2 to compensate a simulated DAC output. It also discusses some limitations of the calibration algorithm and describes how reduce their effects. Finally, Chapter 4 presents a refined step-by-step calibration algorithm.

Chapter 5 presents results obtained by calibration and compensation of a real DAC using the refined calibration system and the compensation techniques described in this thesis.

Chapter 6 sums up the results, presents conclusions on those results, and suggests future work in DAC compensation.

CHAPTER 2

Dynamic Distortion in DACs

This chapter describes several common DAC error specifications and descriptions for static (not frequency dependent) DAC errors. It then explores some dynamic (frequency dependent) error mechanisms present in DACs and develops a DAC model including these errors. Finally this chapter provides a description of an error compensation scheme based on the developed DAC model.

2.1 Descriptions of DAC Distortion

Most specifications of DAC errors describe static errors (those not dependent on the frequency components of the DAC generated signal). Although useful when describing DAC linearity for low frequency output signals, these are not sufficient to describe errors present in higher frequency output signals. This section describes several common static and dynamic error specifications for DACs.

2.1.1 Static Error Descriptions

Before describing DAC errors it is first necessary to discuss the ideal DAC output. The ideal linear DAC output is a constant (β) plus a proportionality (γ) times the input code value. This thesis assumes the constant and proportionality factors (β and γ) for the DAC in question are known.

$$v_o(i) = \beta + \gamma i \tag{2.1}$$

In (2.1) the output voltage $v_o(i)$ has units of volts, β has units of volts, input code i has units of Least Significant Bits (LSBs) and γ has units of volts/LSB. Actual (non-ideal) DACs have errors inherent in their architecture. A common model for these errors is the static model where each DAC output has a constant error value, $e(i)$, associated with it.

$$v_o(i) = \beta + \gamma i + e(i) \tag{2.2}$$

Static errors may be introduced by problems in the DAC architecture. One common cause is unequal current mirrors present in a current summing stage in the DAC. Two common specifications of DAC error are Integral Non-Linearity (INL) and Differential Non-Linearity (DNL) [1, 2]. INL is the maximum deviation over all inputs from the ideal curve measured in (LSBs).

$$\text{INL} = \max_i \{e(i)\}/\gamma \tag{2.3}$$

DNL is the maximum change in error over all inputs measured in LSBs.

$$\text{DNL} = \max_i \{e(i+1) - e(i)\}/\gamma \tag{2.4}$$

These specifications only have meaning when describing DAC error at DC, but are adequate descriptions for the DAC model presented thus far. DAC static error is commonly compensated by laser trimming and other hardware techniques that remove the source of these errors. The static model is not sufficient to describe the behavior of the device as input code i is changed from one value to the next. The next section discusses a DAC model that includes dynamic errors.

2.1.2 Dynamic Error in DACs

Several specifications are used to describe DAC dynamic error. Many of these describe the transient response at the DAC output during the transition between inputs. Some examples of this specification type include settling time, slew rate, and switching time [1]. These effects are typically dominated by the perfor-

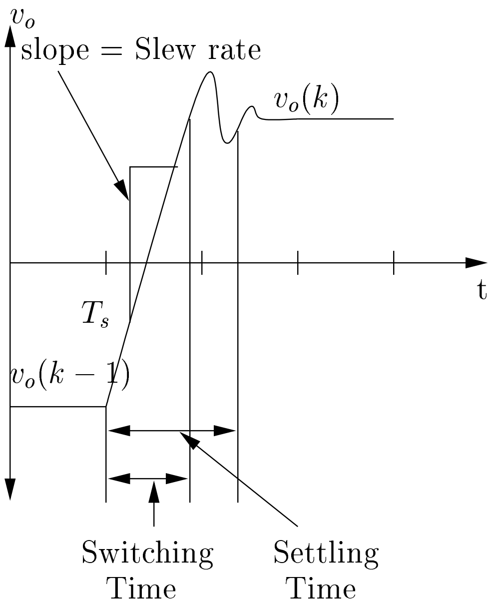

Figure 2.1: Dynamic Specification Definitions

mance of the DAC output stage and are discussed in more detail in Section 2.3.1 and Section 2.3.2. Settling time is the time required to reach and remain within a percentage of the final value. Slew rate describes the maximum voltage rate of change of the DAC output stage. Switching time is the DAC's required time to change output level without settling. Figure 2.1 shows how these specifications are defined as the output changes. All these specifications guarantee some dynamic effects on the output pulse shape to remain within certain bounds. They do not provide information about nonlinear DAC distortion in the frequency domain and may produce no spectral distortion at all. For receiver applications a reconstruction filter is included in the description of DAC performance. A DAC error model with the reconstruction filter is shown in Figure 2.2.

A frequency domain representation of the error becomes more useful for describing distortion mechanisms that produce spurious responses at the output of the reconstruction filter. Some error specifications based on the frequency response include Spurious Free Dynamic Range (SFDR), intermodulation distortion, and

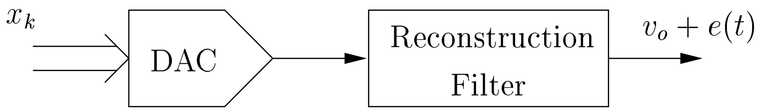

Figure 2.2: Dynamic DAC model with reconstruction filter.

total harmonic distortion (THD). For a digitized sine wave input signal, SFDR is the power ratio (in dB) between the fundamental component of the output signal and the largest spurious component. Figure 2.3 shows a typical measured DAC output spectrum, and illustrates the measurement of SFDR. Intermodulation distortion is the ratio (in dB) of the RMS sum of intermodulation terms created by a two tone input divided by the sum of the RMS amplitudes for the fundamental terms. Total harmonic distortion is the ratio of the first few harmonics' RMS values to the signal RMS [2]. These provide insight about the distortion present at the output of the reconstruction filter. In order to develop compensation techniques for dynamic errors, a DAC model must be adopted which reflects measurable dynamic error mechanisms.

2.2 A Model for Dynamic Distortion

For an ideal DAC, the spectrum at the reconstruction filter output is linearly related to the discrete-time input spectrum. An actual DAC output is corrupted by distortion. In Figure 2.2, the distortion waveform $e(t)$ is a result of the nonlinear behavior of the DAC. We assume that $v_o(t) + e(t)$ can be modeled by corrupting the input sequence to the DAC, and replacing the DAC with an ideal device. That is, a non-ideal DAC with input x_k is equivalent to considering an ideal DAC driven by the corrupted sequence $x_k + \epsilon_k$. The waveform $e(t)$ (and hence the sequence ϵ_k)

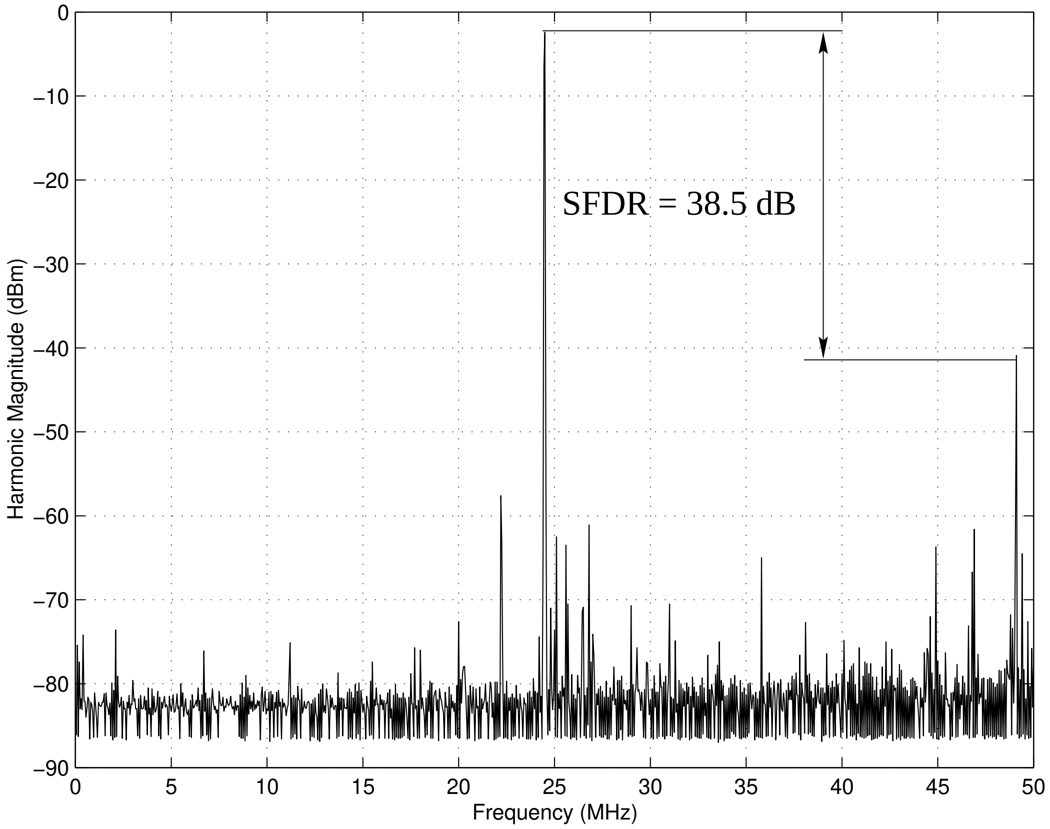

Figure 2.3: A typical SFDR measurement for a DAC driven by a digitized sinusoidal signal at 24.4 MHz.

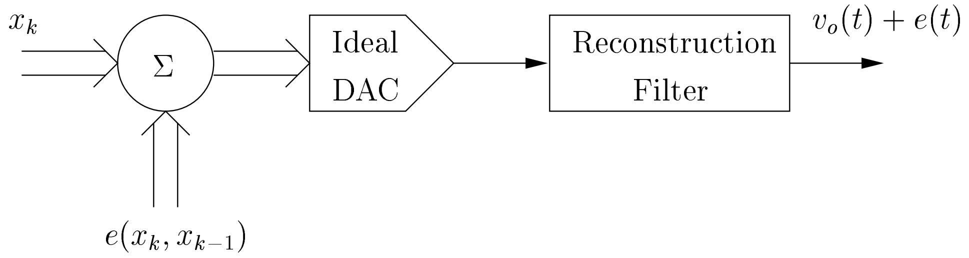

Figure 2.4: Dynamic DAC model with error introduced at the input.

is a function of the sample sequence driving the DAC.

$$\epsilon_k = e(x_k, x_{k-1}, x_{k-2}, \cdots) \tag{2.5}$$

While the above model is fairly general, it should be noted that error mechanisms not related to the input sequence (power supply variations, clock jitter, etc.) are not included. For most DACs, the error introduced by the DAC is primarily a function of x_k and x_{k-1}. Transient effects from the samples $x_{k-2}, x_{k-3}, \cdots$ are small second order effects. To obtain computationally feasible calibration and compensation procedures, we assume that ϵ_k can be modeled as a function of x_k and x_{k-1} only.

$$\epsilon_k = e(x_k, x_{k-1}) \tag{2.6}$$

A block diagram of this DAC model is shown in Figure 2.4. The DAC output for this model is given in equation (2.7).

$$v_o(x_k, x_{k-1}) = \beta + \gamma(x_k + e(x_k, x_{k-1})) \tag{2.7}$$

This dynamic DAC model is capable of modeling several effects seen in actual DACs that the static model cannot. The following section describes some of the distortion mechanisms this system is capable of modeling.

2.3 Examples of Dynamic DAC Distortion Mechanisms

DAC dynamic effects have a variety of causes. Two of these causes are the slew-rate of the DAC output stage and internal effects during bit transition (e.g. switching of internal current sources). Both of these distortion mechanisms provide no static error measurement, i.e. these produce zero INL and DNL. Static calibration and compensation techniques cannot correct for these errors.

2.3.1 Slew-Rate Effects

Slew-rate effects produce distortion by limiting the output rate of change. They are commonly caused by limitations in the output amplifier stage of the DAC [1]. This limitation causes the DAC output to ramp from one output to the next as shown in Figure 2.5. Observation of Figure 2.5 reveals that this distortion mechanism has zero INL and DNL because the output settles to the desired value. This mechanism is capable of introducing harmonic distortion in the DAC output. To illustrate this effect, an 8-bit slew-limited DAC was simulated with zero integral and differential error. The simulated DAC has an output range of $\pm$ 1V and an output amplifier slew rate of 4 V per sample period T_s. The simulated DAC output is over-sampled by a factor of 32 to produce the transition points. For an input sequence creating a 9.8 MHz output tone, the simulated DAC output spectrum is shown in Figure 2.6. The DAC provides an SFDR of approximately 42 dB for this relatively low frequency input signal.

This error is created only when the output voltage changes as a result of an input code change. Assuming the slew limit allows the output to reach the steady state value, this error is a function of the current input and previous input. Therefore the model in Figure 2.4 is one possible model for this type of distortion. Amplitude dependent transition times caused by the slew rate limit create harmonic distortion observable in the DAC output spectrum. This does not

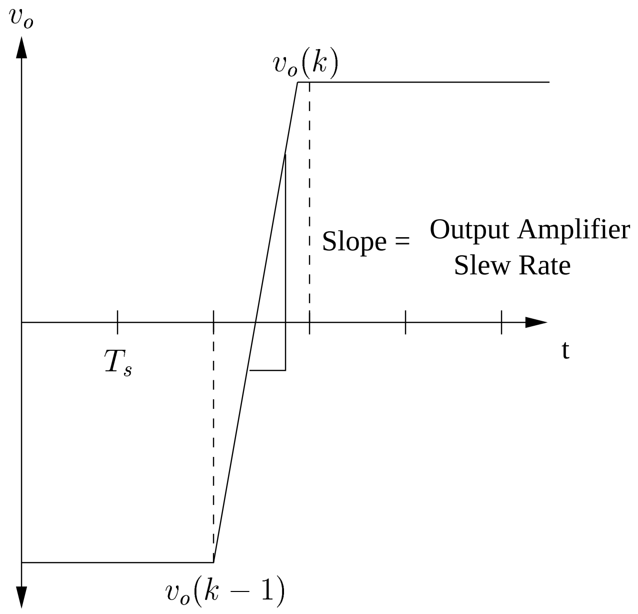

Figure 2.5: DAC slew limiting example output.

introduce any INL or DNL component, however, because the output settles to a static value that is independent of slewing.

2.3.2 Bit-Transition and Glitch Effects

Bit-transition effects, including glitch effects, are those caused by internal switching within the DAC when changing from one state to the next. They tend to be most significant when many input bits are being changed simultaneously. A model of this type of distortion is a constant error produced whenever a particular bit is switched from zero to one and the negative of that error produced whenever the same bit is switched from one to zero. An equation for this error model is shown in equation (2.8) for an m-bit DAC with σ_q error on bit q.

$$e(x_k, x_{k-1}) = \sum_{q=1}^{m} \sigma_q (b_q(x_k) - b_q(x_{k-1})) \tag{2.8}$$

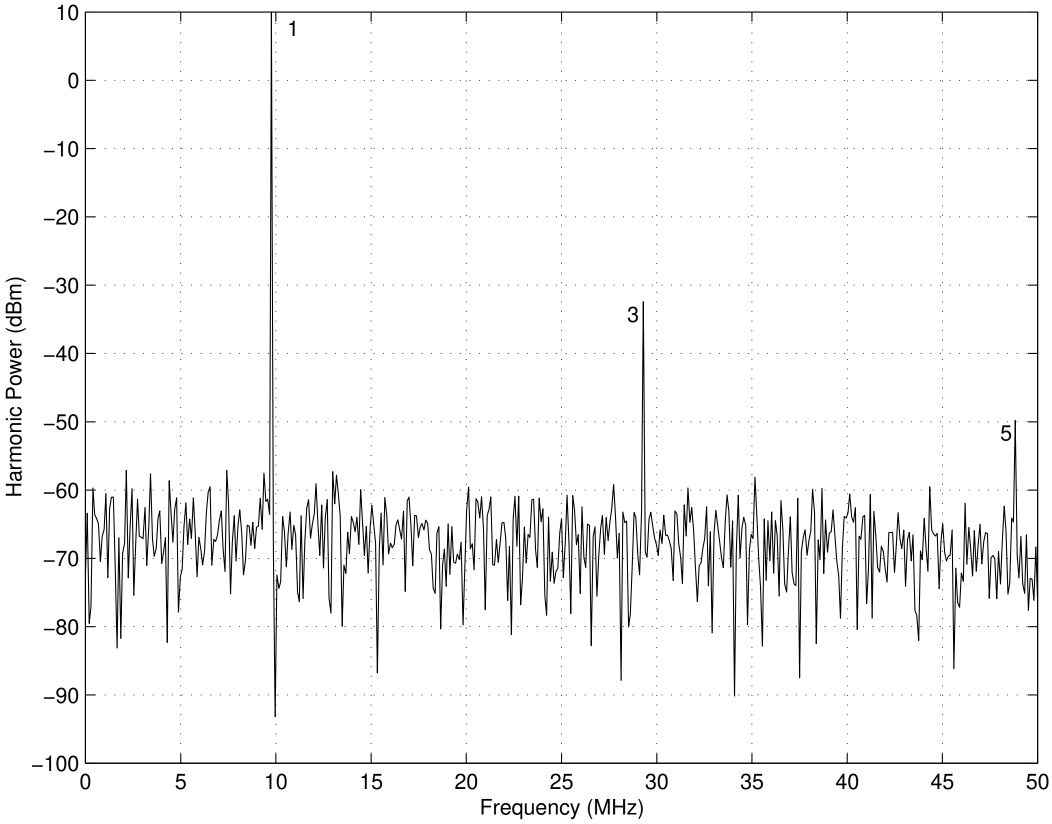

Figure 2.6: Simulated slew limited DAC output spectrum.

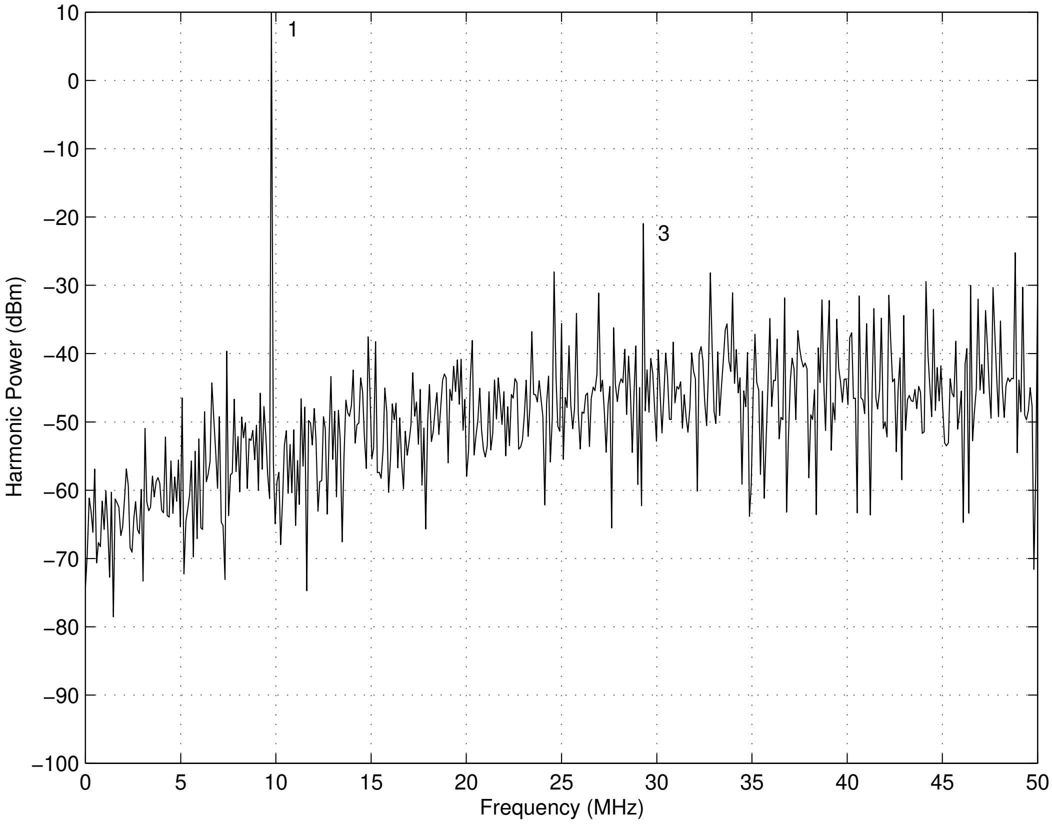

Figure 2.7: Simulated DAC with bit-transition errors.

In equation (2.8) $b_q(x_k)$ is the value (one or zero) of bit q for input x_k, and $b_q(x_{k-1})$ is the value of bit q for the previous input. This error type produces harmonic distortion, but cannot be modeled by a function of the current input alone. Like the slew-rate limits, this effect is not measured at steady state, and is not reflected by the DAC INL or DNL. An 8-bit DAC with this distortion model is simulated and a sinusoidal input creating a 9.8 MHz output tone is applied to the DAC input. The simulated output spectrum for this system is shown in Figure 2.7.

Glitching is caused by internal DAC switches with unequal rise and fall times [1]. This causes the input bits to appear to be all ones or zeros until switching is complete, making the DAC output temporarily jump to a maximum or minimum voltage. This produces harmonic distortion dependent on the current and previous

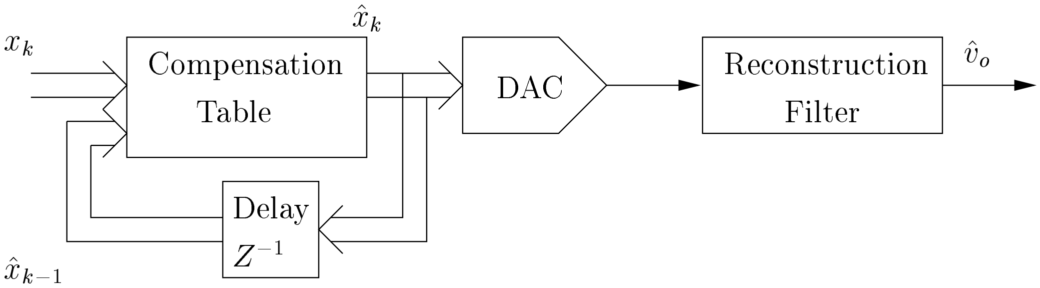

Figure 2.8: Dynamic DAC error compensation.

input, but does not have any static INL or DNL errors because this dynamic effect settles out during a static measurement.

2.4 Compensation of Dynamic Distortion

If the errors in model shown in Figure 2.4 are known for all combinations of current and previous input, then the DAC output for any combination of current and previous input may be calculated. The desired output may be determined using the ideal DAC equation. The input that produces the output closest to the desired DAC output for the given previous input can be determined using the actual DAC input-output characteristic defined by equation (2.7). The DAC input that produces the output value closest to the desired output is then the compensated input to the DAC, $\hat{x}_k$. An implementation utilizes the desired DAC input and the previous DAC input to index a compensation table to determine the DAC input that will produce the desired output level. A block diagram is shown in Figure 2.8. Using this compensation model, the next step is to calculate the values in the compensation table.

2.4.1 Construction of the Compensation Table Entries

Compensation table construction involves determining the actual input-output characteristic of the non-ideal DAC and calculating the most suited input for a given desired output and previous input. The steps involved in creating this table are listed below:

1. Measure the error characteristic (i.e. $e(x_k, x_{k-1})$) for all combinations of x_k and x_{k-1}.

2. Given a previous input to the DAC (x_{k-1}), calculate the DAC output characteristic as a function of all possible current input values, $\hat{x}_k$ based on the model shown in Figure 2.4.

$$v_o(\hat{x}_k, x_{k-1}) = \beta + \gamma(\hat{x}_k + e(\hat{x_k}, x_{k-1})) \tag{2.9}$$

3. Calculate the desired DAC output for the given input, x_k, using the ideal DAC equation (equation (2.1)).

$$v_{od} = \beta + \gamma x_k \tag{2.10}$$

4. Find the output value, v_o, in equation (2.9) that minimizes $(v_{od} - v_o)^2$. The $\hat{x}_k$ that produces the minimum error v_o is the table entry for the given previous DAC input and current desired DAC input.

To show the effects of compensation, the simulated DAC with known error values described in Section 2.3.2 is compensated using this scheme. The compensated spectrum shown in Figure 2.9 shows superior performance to the uncompensated spectrum in Figure 2.7 with a compensated SFDR of 54.0 dB versus the uncompensated SFDR of 31.32 dB. Based on this DAC model and compensation scheme

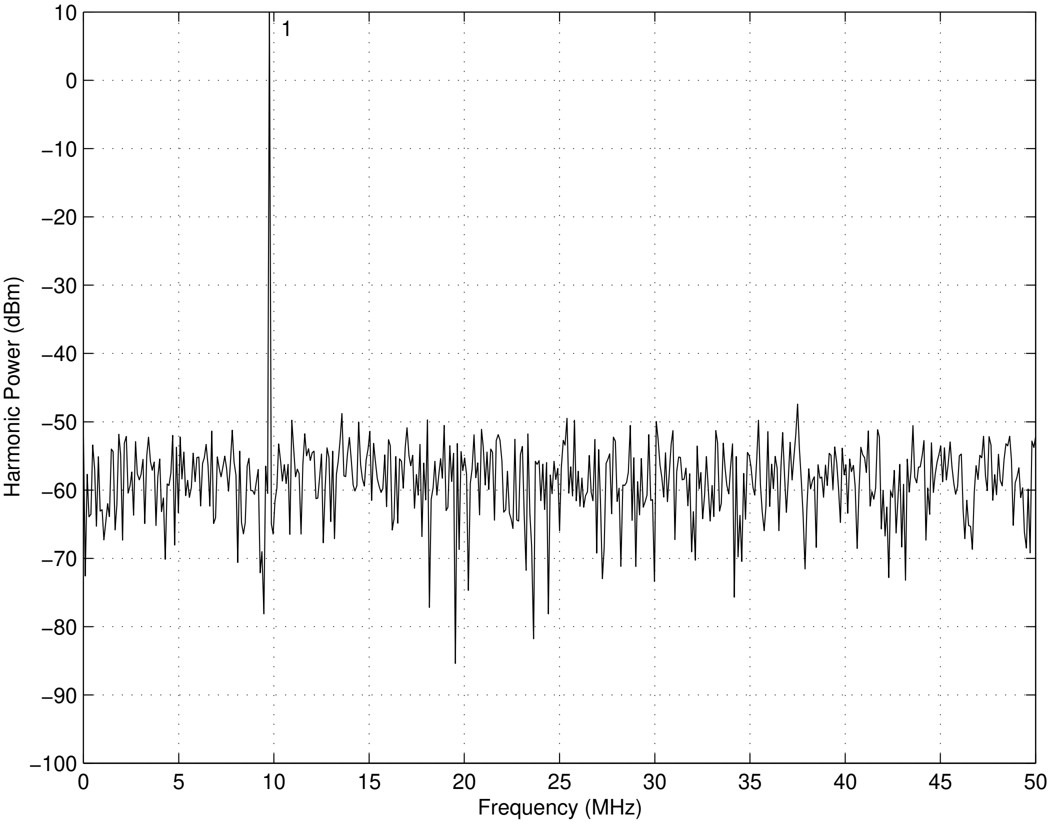

Figure 2.9: Compensated spectrum for simulated DAC with bit transition errors.

the next requirement is to develop a method for measuring the converter errors $e(x_k, x_{k-1})$.

CHAPTER 3

DAC Dynamic Error Estimation

Digital to Analog Converters (DACs) transform discrete-time codewords into real-time voltage levels that change only at clock specified times. The ideal DAC output characteristic produces evenly spaced voltage outputs such that the ratio of voltage change to input least significant bit (LSB) change remains a constant. A real (non-ideal) DAC does not have this characteristic because of error mechanisms present in the architecture. This chapter develops a method for measuring non-ideal DAC dynamic errors.

3.1 A Lowpass Measurement Approach

The first step in measuring the DAC characteristic is quantization of DAC error data so that it may be manipulated with a computer. By driving the DAC using predetermined sequences and measuring the output waveform, DAC distortion characteristics may be estimated. A possible scheme is to sample the output of the DAC using a higher performance (higher resolution at the same speed) ADC as illustrated in Figure 3.1. Unfortunately, an Analog to Digital Converter (ADC) of equal or better performance to the DAC is generally not available. In practice,

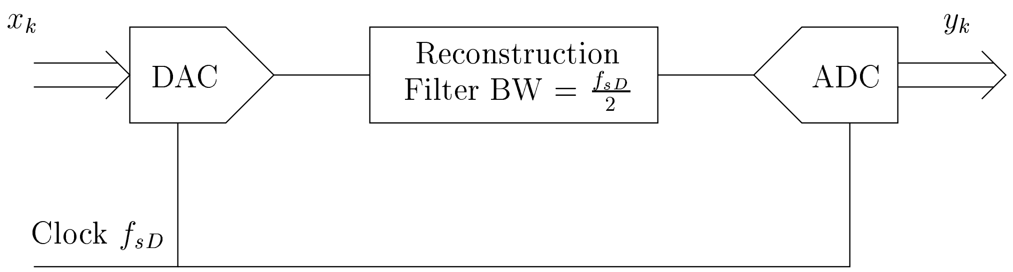

Figure 3.1: DAC calibration system using high performance ADC.

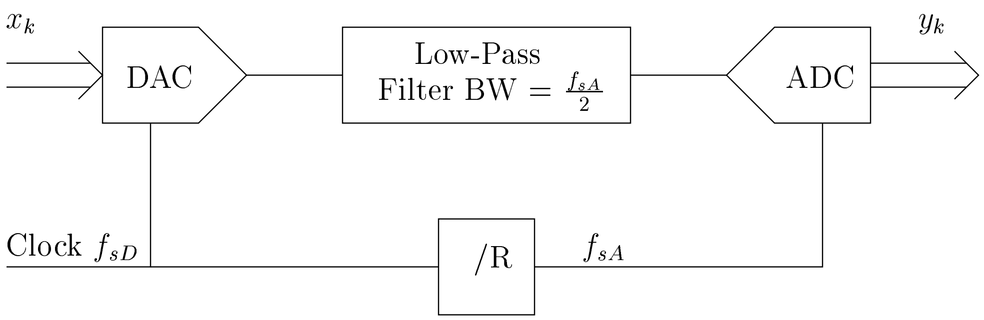

Figure 3.2: DAC calibration system using high resolution ADC at a reduced sampling rate.

ADC errors tend to dominate DAC errors for devices with comparable sample rates. A system that utilizes a high performance ADC that operates at a slower clock rate is a possible solution. The calibration system described in this thesis is shown in Figure 3.2. This system clocks the ADC at the DAC rate divided by R. A lowpass filter is added to the output of the DAC to remove frequencies outside the Nyquist band of the ADC. The input impedance of this filter must be carefully designed so that it provides a constant load to the DAC for all frequencies to prevent large reflections loading the DAC output for filtered signal terms. This allows the user to implement a slower ADC with linearity superior to the DAC. This superior ADC may then measure DAC output errors that appear at frequencies within the filter passband. To utilize this system, a method of measuring the DAC error characteristic from the limited observed spectrum is necessary.

3.2 Approximation of $e(x_k, x_{k-1})$

The output of the calibration system in Figure 3.2 provides a limited amount of DAC output spectrum information. The error of the converter, as modeled by Figure 2.4 is a non-linear function of the current and previous input to the DAC. Error is measured using periodic input sequences to allow Discrete Fourier Transform (DFT) analysis of the ADC output codes to identify DAC distortion terms. The DAC input signals are structured so that the DAC is excited over the entire DAC Nyquist bandwidth with significant distortion terms within the observation band. These distortion terms are then used to determine a set of parameters which estimate the error function $e(x_k, x_{k-1})$. To implement the calibration, we address several technical issues:

1. The relationship between the DFT of the output sequences y_k and the input sequence x_k (or the error sequence $e(x_k, x_{k-1})$) must be determined (Section 3.2.1).

2. A linear technique to establish the gain and phase shift associated with the measurement system must be developed (Section 3.2.2).

3. The error function $e(x_k, x_{k-1})$ must be written in terms of a finite number of parameters (Section 3.2.3).

4. A procedure to solve for the error function unknown parameters described in step 3 must be developed (Section 3.2.4).

For calibration the DAC input is driven with periodic input sequences that repeat every N_{sD} samples. Let $\overline{x}$ denote a vector containing one period of the input sequence, and let the kth element of that sequence be denoted x_k. Also let the ADC output be a N_{sA} long sequence denoted $\overline{y}$ with the kth element denoted y_k. Finally let $\overline{e}$ denote the DAC error vector associated with the input sequence, $\overline{x}$.

3.2.1 DFT Relationship between $\overline{x}$ (or $\overline{e}$) and $\overline{y}$

In order to compensate the converter by altering the input, some relationships between $\overline{x}$ (or $\overline{e}$) and $\overline{y}$ must be determined. The first important relationship is the frequency bin mapping between the DFTs of $\overline{x}$ (or $\overline{e}$) and $\overline{y}$. If f_{sD} is the

DAC output sampling frequency and N_{sD} is the input sequence length, then we can calculate the output frequency, f_{oD}, for a given DAC input DFT bin number ν_D.

$$f_{oD} = \nu_D \frac{f_{sD}}{N_{sD}} \tag{3.1}$$

Now given the input frequency to the ADC it is possible to calculate the ADC DFT bin number, ν_A.

$$\nu_A = f_{oD} \frac{N_{sA}}{f_{sA}}. \tag{3.2}$$

In equation (3.2), N_{sA} is the number of samples taken from the ADC and f_{sA} is the sampling frequency of the ADC. The system described in this thesis (Figure 3.2) sets the ADC sampling rate to the DAC sampling rate divided by a decimation factor, R.

$$f_{sA} = \frac{f_{sD}}{R} \tag{3.3}$$

Now if we substitute equations (3.3) and (3.1) into equation (3.2) a mapping between the input sequence bin numbers and the output sample bin numbers is obtained.

$$\nu_A = \nu_D \frac{N_{sA} R}{N_{sD}} \tag{3.4}$$

This equation determines the bin mapping factor, k, for this system:

$$k = \frac{N_{sA} R}{N_{sD}} \tag{3.5}$$

From equation (3.5) it is apparent that for an integer decimation factor, R, the mapping factor will be an integer when the number of samples taken from the ADC times the decimation factor is a multiple of the input sequence length. This is required for the assumption that the ADC output sequence repeats every N_{sA} samples (making the DFT of the sequence meaningful). Further discussion in this thesis assumes the mapping factor, k, is an integer.

3.2.2 Linear Characterization of System Gains and Phases

In order to perform an accurate calibration the effects of the calibration system on the system output that are not related to DAC error must be removed. The most significant of these effects are the phase and amplitude changes introduced by the low-pass filter and the signal gains and pipeline delays associated with the DAC and ADC. The removal of gain and phase change effects is accomplished by doing a linear characterization of the system. This linear characterization produces a vector $\overline{L}$ such that L_i is the gain and phase associated with input bin i. Let ζ be the set of all possible input sequence frequency bins that map into ADC bins within the calibration region (i.e. within the passband of the low-pass filter). The system gain and phase associated with each of these bins may be determined by setting the DAC input to a single cosine tone sequence for every bin in ζ and measuring the gain and phase shift for that tone at the ADC output. Now define $\tilde{T}$ as a Fourier Transform matrix [7] such that $\tilde{T}\overline{x}$ produces the Fourier Transform of $\overline{x}$. The elements of $\tilde{T}$ are defined by the complex quantity W_N calculated for the sequence length N.

$$W_N = e^{-j\frac{2\pi}{N}} \tag{3.6}$$

The Fourier Transform matrix is defined by equation (3.7).

$$\tilde{T} = \begin{bmatrix} {W_N}^0 & {W_N}^0 & {W_N}^0 & \dots & {W_N}^0 \\ {W_N}^0 & {W_N}^1 & {W_N}^2 & \dots & {W_N}^{N-1} \\ {W_N}^0 & {W_N}^2 & {W_N}^4 & \dots & {W_N}^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ {W_N}^0 & {W_N}^{N-1} & {W_N}^{2(N-1)} & \dots & {W_N}^{(N-1)(N-1)} \end{bmatrix}. \tag{3.7}$$

By observation of $\tilde{T}$ it is apparent that the rows are given by the Fourier Transform coefficients for the corresponding DFT bins. Now let $\overline{T}_i$ denote the ith row of $\tilde{T}$, such that $\overline{T}_i\overline{x}$ is the ith DFT harmonic value of $\overline{x}$. If $\overline{p}$ is a vector of frequency bins then let $\tilde{T}_{\overline{p}}$ be a matrix containing the rows of $\tilde{T}$ specified by the vector $\overline{p}$. Also let $\tilde{T}^A$ be the T matrix defined for the length of the ADC output sequence and let $\tilde{T}^D$ be the $\tilde{T}$ matrix defined for the DAC input sequence length. The $\overline{L}$ vector is calculated by dividing the ADC output harmonic value by the DAC input harmonic value. In order to match the proper harmonics the row of the ADC transform matrix is determined by the mapping factor, k times the DAC harmonic bin number. For DAC bin number, i, the ADC output bin number that harmonic appears on is ki. To determine a given L_i, a DAC input sequence, $\overline{x}_i$, is generated as a single sinewave tone at bin i. The ADC output sequence for the DAC input is retrieved and denoted $\overline{y}_i$. L_i is then the ADC output DFT evaluated at the appropriate bin, ki, divided by the DFT of the DAC input at bin i as shown in equation (3.8).

$$L_i = \frac{\overline{T}^A_{ki}\overline{y}_i}{\overline{T}^D_i\overline{x}_i} \tag{3.8}$$

With the system gain and phase behavior modeled, it becomes possible to develop basis function estimation of DAC distortion procedures. Note that the above linear characteristic may be used to calculate a component of the DAC error using the ADC outputs, since x_k and $e(x_k, x_{k-1})$ are passed through the same system. For a non-fundamental tone bin i, the DAC error component is given in equation (3.9) for the system shown in Figure 3.3.

$$\overline{T}^D_i\overline{e} = \frac{1}{L_i}\overline{T}^A_{ki}\overline{y} \tag{3.9}$$

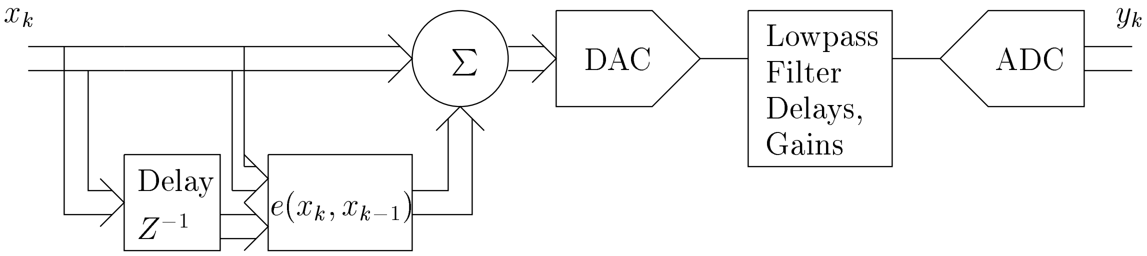

Figure 3.3: Calibration system with error at DAC input.

3.2.3 A Basis Function Representation of $e(x_k, x_{k-1})$

In order to estimate the unknown function $e(x_k, x_{k-1})$, a representation in terms of a finite number of unknown coefficients is useful. Basis function estimation of a non-linear system involves evaluating a set of non-linear functions (the basis functions) of the system inputs and multiplying each basis function by its respective weight. In this thesis a set of N_b basis functions is used to approximate the DAC converter error function, $e(x_k, x_{k-1})$. c_j is the basis function weight associated with basis function j. The error estimate is formed by multiplying the basis function outputs, $b_j(x_k, x_{k-1})$, by their respective weights as shown in equation (3.10).

$$\hat{e}(x_k, x_{k-1}) = \sum_{j=1}^{N_b} c_j b_j(x_k, x_{k-1}) \tag{3.10}$$

Let $\overline{c}$ be the N_b by 1 vector of weights associated with the N_b basis functions. Usually the basis function weights are calculated using a least-mean-square (LMS) approximation to an over determined system [8]. Let $\overline{B}(x_k, x_{k-1})$ be the 1 by N_b vector of basis function outputs for a given input and previous input as shown in equation (3.11).

$$\overline{B}(x_k, x_{k-1}) = [b_1(x_k, x_{k-1}) \; b_2(x_k, x_{k-1}) \; \cdots \; b_{N_b}(x_k, x_{k-1})] \tag{3.11}$$

The estimate of the error waveform for a given input and previous input, $\hat{e}(x_k, x_{k-1})$ is then the basis function vector times the weight vector as shown in equation (3.12).

$$\hat{e}(x_k, x_{k-1}) = \overline{B}(x_k, x_{k-1})\overline{c} \tag{3.12}$$

For a set of inputs $\overline{x}$, let $\overline{x}_{-1}$ be a circularly delayed vector of inputs as shown in equation (3.13).

$$\overline{x}_{-1} = [x_{N_{sD}} \; x_1 \; x_2 \; \cdots \; x_{N_{sD}-1}]^T \tag{3.13}$$

Given the input vectors $\overline{x}$ and $\overline{x}_{-1}$ a N_{sD} by N_b basis function output matrix $\tilde{B}$ can be calculated. The ith row and kth column of $\tilde{B}$ is the kth basis function evaluated at the ith value of $\overline{x}$ and $\overline{x}_{-1}$. For a given $\tilde{B}$ the estimate of the errors associated with the input sequence $\overline{x}$, $\hat{E}$, is $\tilde{B}$ times the weight vector, $\overline{c}$, as shown in equation(3.14).

$$\hat{E} = \tilde{B}\overline{c} \tag{3.14}$$

With this notation and the frequency domain relations discussed in Section 3.2.1 and Section 3.2.2, it is possible to determine the basis functions weights, $\overline{c}$.

3.2.4 Estimation of Basis Function Weights

The first step in the calibration procedure is the generation of a DAC input test sequence, $\overline{x_k}$, that is N_{sD} long. The input sequences described in this thesis consist of five frequency tones with random phases. With only a limited portion of the spectrum available, test tones are chosen carefully in order to produce low frequency error terms the ADC can measure. Further discussion of how these test tones are selected is given in Section 3.3. The generated input sequence is applied to the DAC input. An output sequence, $\overline{y}$, of length N_{sA} is retrieved from the ADC output. The basis function output matrix, $\tilde{B}$, is evaluated for every input and previous input, $\overline{x}$, $\overline{x}_{-1}$. A set of bins is then chosen to examine the error within the calibration region; these are denoted $\overline{p}$ and are elements of ζ. Unless otherwise designated, the simulations in this thesis use all bins in ζ except for the locations of the test frequencies modulo $\frac{N_s D}{R}$ so that error terms, not signal terms, are estimated. According to equation (3.14) the error estimate $\hat{E}$ is $\tilde{B}\overline{c}$. To compute $\overline{c}$, a least-mean-squares (LMS) fit [8] is performed in the frequency domain to make the frequency response of $\hat{E}_k$ a best fit approximation to the frequency response of the DAC output for non-fundamental tones. From equation (3.9) we know the harmonics of $\overline{e}$ can be observed. We want to find $\hat{E}$ by selecting $\overline{c}$ so that the harmonics of $\overline{e}$ are closely approximated by the harmonics of $\hat{E}$. The approximation of a given bin harmonic, i, will have some residual error left by the approximation, denoted ϵ_i as shown in equation (3.15).

$$\overline{T}_i^D \hat{E} = \overline{T}_i^D \overline{e} + \epsilon_i \tag{3.15}$$

Substituting equation (3.14) and equation (3.9) into equation (3.15) yields equation (3.16).

$$\overline{T}_i^D \tilde{B}\overline{c} = \frac{1}{L_i} \overline{T}_{ki}^A \overline{y} + \epsilon_i \tag{3.16}$$

Multiplying both sides of the equation by L_i eliminates the need to compute an inverse when using matrix notation and emphasizes error terms that are less attenuated by the filter. This yields equation (3.17).

$$L_i \overline{T}_i^D \tilde{B}\overline{c} = \overline{T}_{ki}^A \overline{y} + \epsilon_i' \tag{3.17}$$

Let $\overline{L}_{\overline{p}}$ be a vector of the $\overline{p}$ elements of $\overline{L}$. Next let $\tilde{L}_{\overline{p}}$ be a diagonal matrix with diagonal $\overline{L}_{\overline{p}}$. Equation (3.17) can then be expressed in the matrix form given in equation (3.18).

$$\tilde{L}_{\overline{p}}\tilde{T}_{\overline{p}}^{D}\tilde{B}\overline{c} = \tilde{T}_{k\overline{p}}^{A}\overline{y} + \overline{\epsilon}' \tag{3.18}$$

When implementing, $\tilde{T}_{\overline{p}}^{D}$ and $\tilde{B}$ can be large matrices. To reduce memory requirements $\tilde{G}$ is formed as shown in equation (3.19).

$$\tilde{G} = \tilde{T}_{\overline{p}}^{D}\tilde{B} \tag{3.19}$$

$\tilde{G}$ is a matrix whose i, j element is given by the p_i DFT bin of $b_j(x_k, x_{k-1})$. Typically $\tilde{G}$ is constructed by performing a DFT on $\tilde{B}$ and removing the desired terms. Using $\tilde{G}$, the basis function frequency domain calibration matrix, $\hat{B}$ is calculated as shown in equation (3.20). The ith row of this matrix is the ith row of $\tilde{G}$ scaled by $\overline{L}_{p_i}$.

$$\hat{B} = \tilde{L}_{\overline{p}}\tilde{G} \equiv \tilde{L}_{\overline{p}}\tilde{T}_{\overline{p}}^{D}\tilde{B} \tag{3.20}$$

The $\overline{y}$ DFT is calculated using the T matrix for the ADC output.

$$\overline{Y} = \tilde{T}_{k\overline{p}}^{A}\overline{y} \tag{3.21}$$

With the notation introduced by equations (3.20) and (3.21) equation (3.18) becomes equation (3.22).

$$\hat{B}\overline{c} = \overline{Y} + \overline{\epsilon}' \tag{3.22}$$

To allow a multiple signal test, $\hat{B}$ and $\overline{Y}$ are stored in the storage matrix, $\hat{B}_s$, and the storage vector, $\overline{Y}_s$, respectively. The error estimate must be real to allow compensation. In order to create a real signal that best reproduces the real and imaginary frequency domain responses, the real and imaginary parts of $\hat{B}$ and $\overline{Y}$ are included as separate entries in storage variables, $\hat{B}_s$ and $\overline{Y}_s$. Equation (3.23) shows how the values are stored in $\hat{B}_s$ and equation (3.24) shows how the values are stored in $\overline{Y}_s$ for a set of M test signals where $\hat{B}^m$ represents the $\hat{B}$ matrix for the mth test signal and $\overline{Y}^m$ is the $\overline{Y}$ vector for the mth test signal.

$$\hat{B}_s = \begin{bmatrix} \mathrm{Re}(\hat{B}^1) \\ \mathrm{Im}(\hat{B}^1) \\ \mathrm{Re}(\hat{B}^2) \\ \mathrm{Im}(\hat{B}^2) \\ \vdots \\ \mathrm{Re}(\hat{B}^M) \\ \mathrm{Im}(\hat{B}^M) \end{bmatrix}. \tag{3.23}$$

$$\overline{Y}_s = \begin{bmatrix} \mathrm{Re}(\overline{Y}^1) \\ \mathrm{Im}(\overline{Y}^1) \\ \mathrm{Re}(\overline{Y}^2) \\ \mathrm{Im}(\overline{Y}^2) \\ \vdots \\ \mathrm{Re}(\overline{Y}^M) \\ \mathrm{Im}(\overline{Y}^M) \end{bmatrix}. \tag{3.24}$$

Values are stored until a user defined number of calibration tone sets have been applied to the converter. Once enough test signals have been run, the basis function weight vector, $\overline{c}$, is calculated to minimize $\overline{\epsilon}'^T\overline{\epsilon}'$ using a LMS pseudo-inverse fit [8].

$$\overline{c} = (\hat{B}_s^T\hat{B}_s)^{-1}\hat{B}_s^T\overline{Y}_s \tag{3.25}$$

The determination of $\overline{c}$ allows the calculation of the converter error estimate, $\hat{E}$, for any set of current and previous DAC inputs.

3.3 Selection of the Calibration Frequencies

The selection of calibration frequencies must be done to make significant error terms appear in the calibration region of the system. Since this is a low-pass system it is necessary to create test signals that produce near-baseband error terms. Significant error terms generally appear at harmonic values (multiples of the fundamental tones) and at intermodulation frequencies (sum and difference combinations of fundamental tones and their harmonics). For an even polynomial distortion mechanism, intermodulation error terms appear close to baseband when sets of signal frequencies are chosen close together. For example, consider a DAC characteristic whose transfer function contains an even order term as described in equation (3.26).

$$\overline{y} = \overline{x} + \psi\overline{x}^2 \tag{3.26}$$

ψ is a constant related to the error power in equation (3.26). Significant error terms appear near baseband when $\overline{x}$ is a two-tone signal with frequencies close together. A simulation of this effect is shown in Figure 3.4. Note that all labeled frequency locations are taken modulo f_sD. In this example the DAC characteristic described in equation (3.26) is used to determine the output for a two tone signal with the frequencies spaced two MHz apart. This input signal produces a two MHz intermodulation error term that could be used to calibrate the DAC. Likewise, when the transfer function contains odd powers of $\overline{x}$, error terms appear close to baseband when frequencies are chosen with one tone nearly double the other. An odd characteristic given in equation (3.27) is used to determine the output for a two tone signal with one frequency at double the other plus two MHz (i.e. $f_2 = 2f_1 + 2$).

$$\overline{y} = \overline{x} + \psi\overline{x}^3 \tag{3.27}$$

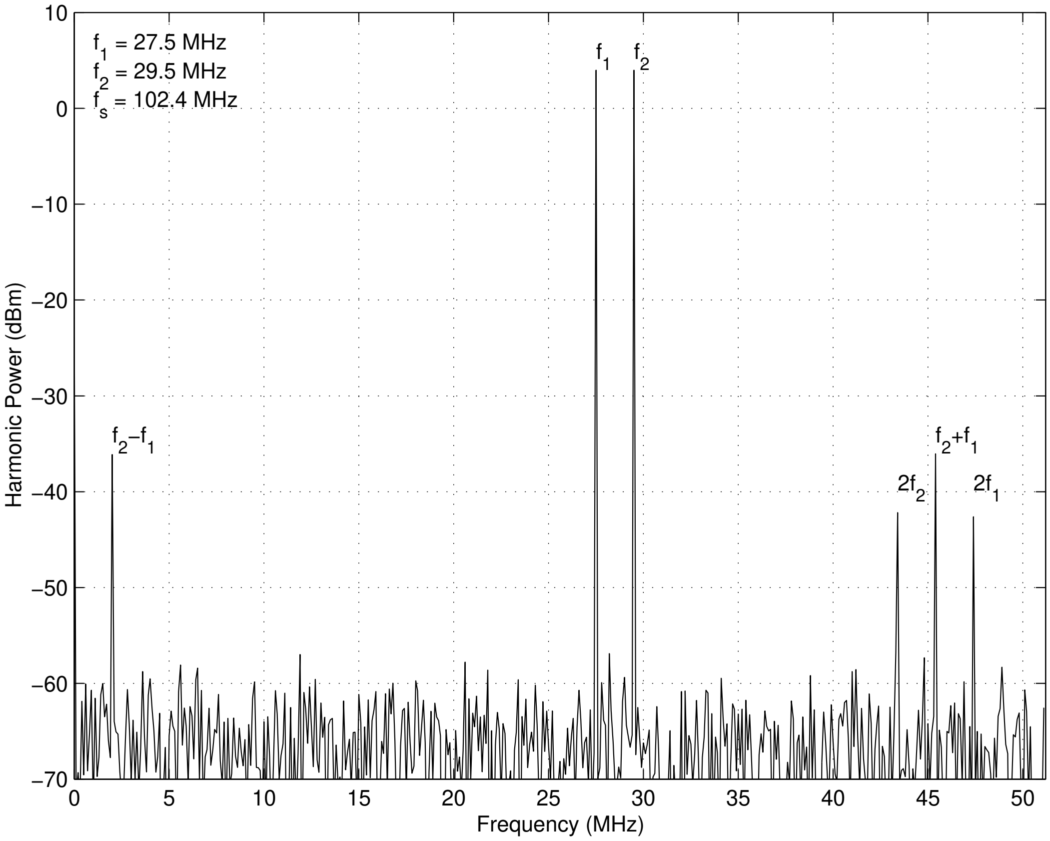

Figure 3.4: Intermodulation error for second order characteristic.

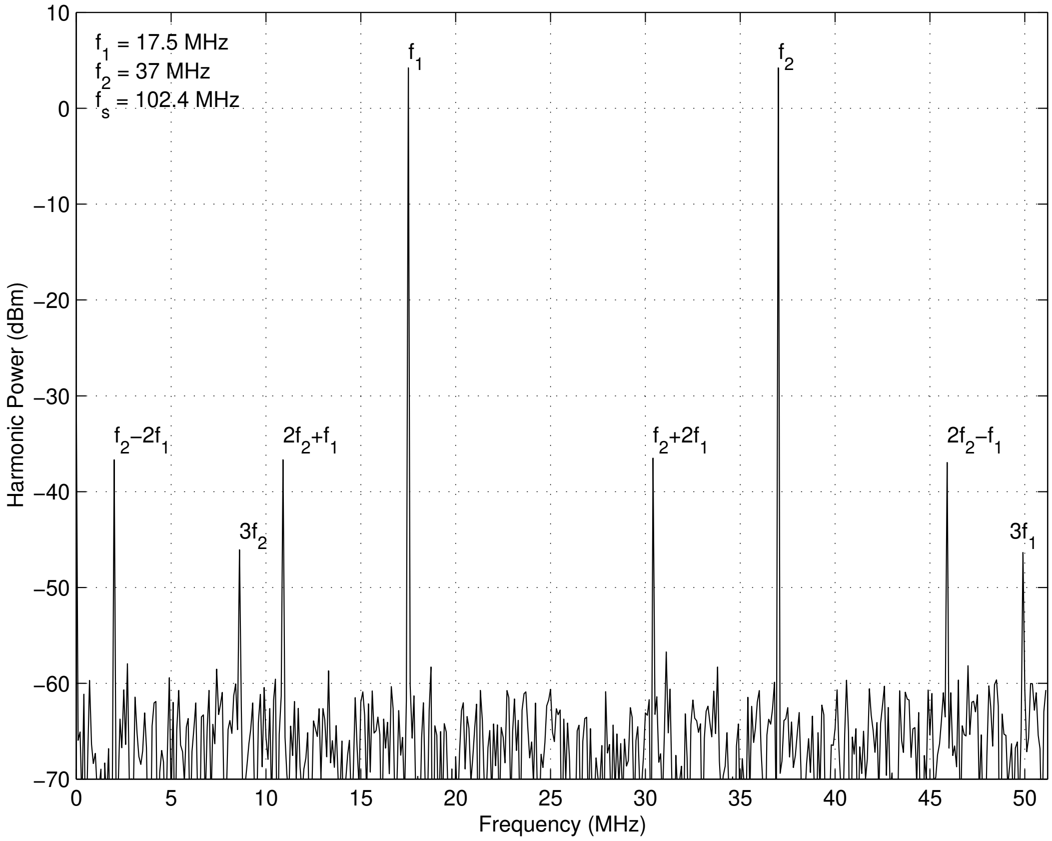

Figure 3.5: Intermodulation error for third order characteristic.

When this system is simulated an error term appears at two MHz, the separation between double the first frequency and the second frequency as shown in Figure 3.5. Therefore, in order to measure both even and odd order error characteristics, test signals must be chosen with some frequencies that are close together and others that are double frequencies. This may be done by either using different two-tone signal sets or a richer multi-tone signal consisting of both close and double tones together. An example of such a signal construction is given in Chapter 4.

CHAPTER 4

Simulated Results (and Refinement of the Calibration)

With a method of measuring the converter error characteristic (as described in Section 3.2) and a compensation method (as described in Section 2.4) available, DAC input signal compensation may be performed to produce improved DAC linearity. This chapter will first describe a simple DAC model with dynamic distortion and discuss results obtained from calibration and compensation of this model. Next this chapter will cover some limitations of this calibration system and describe techniques used to combat these limitations. Section 4.3 then investigates effects of limited calibration bandwidth and discusses methods to improve calibration system performance for highly limited bandwidths. Finally, this chapter develops a refined calibration algorithm utilizing techniques described in Sections 4.2 and 4.3 that provides improved compensation performance.

4.1 A Simple Dynamic DAC Model

A simple method of DAC error modeling is the introduction of polynomial distortions that produce non-linear performance. The ideal DAC described in equation (2.1) has a linear input code to output-voltage relationship. The DAC model described in this section contains additive static and dynamic errors. To introduce static errors this linear model is corrupted by higher order polynomial terms that are functions of the input code. The static error component of this model is given by equation (4.1).

$$v_o(x_k) = \beta + \gamma x_k + \upsilon(\beta + \gamma x_k)^2 + \delta(\beta + \gamma x_k)^3 \tag{4.1}$$

Error Parameter	Value
υ	.003
δ	.0001
η	.0001
κ	.002

Table 4.1: Error parameter table for simulated DAC.

Note that all terms scale x_k by γ and add β before distorting. This is done to make the error constants υ and δ relate to the DAC output voltage value range that is set to be $\pm$ 1V, not the input code value range that is dependent on DAC resolution. One method for introducing DAC dynamic errors is the introduction of polynomial terms that are functions of the difference between the current input and previous input codes. This error mechanism produces greater errors as the rate of change in the input becomes large. A DAC model of this type with both static and dynamic errors is given in equation (4.2).

$$\begin{aligned} v_o(x_k) = \beta + \gamma x_k + \upsilon(\beta + \gamma x_k)^2 + \delta(\beta + \gamma x_k)^3 + \\ \eta(\beta + \gamma(x_k - x_{k-1}))^2 + \kappa(\beta + \gamma(x_k - x_{k-1}))^3 \end{aligned} \tag{4.2}$$

A simulation using this DAC model with error parameters given in Table (4.1) is performed. Figure (4.1) shows the simulated DAC output spectrum using an uncompensated input to a 12 bit DAC. The DAC error is measured using a simulation of the lowpass calibration system (described in detail in the following section) and error estimation techniques described in Chapter 3. A compensation of this DAC using the measured error and the techniques described in Section 2.4 is performed resulting in the compensated spectrum of Figure 4.2. The DAC output spectrum for a fullband compensated input shows greatly improved performance (74.3 dB SFDR vs. 62.4dB SFDR for uncompensated) over the entire spectrum. This re-

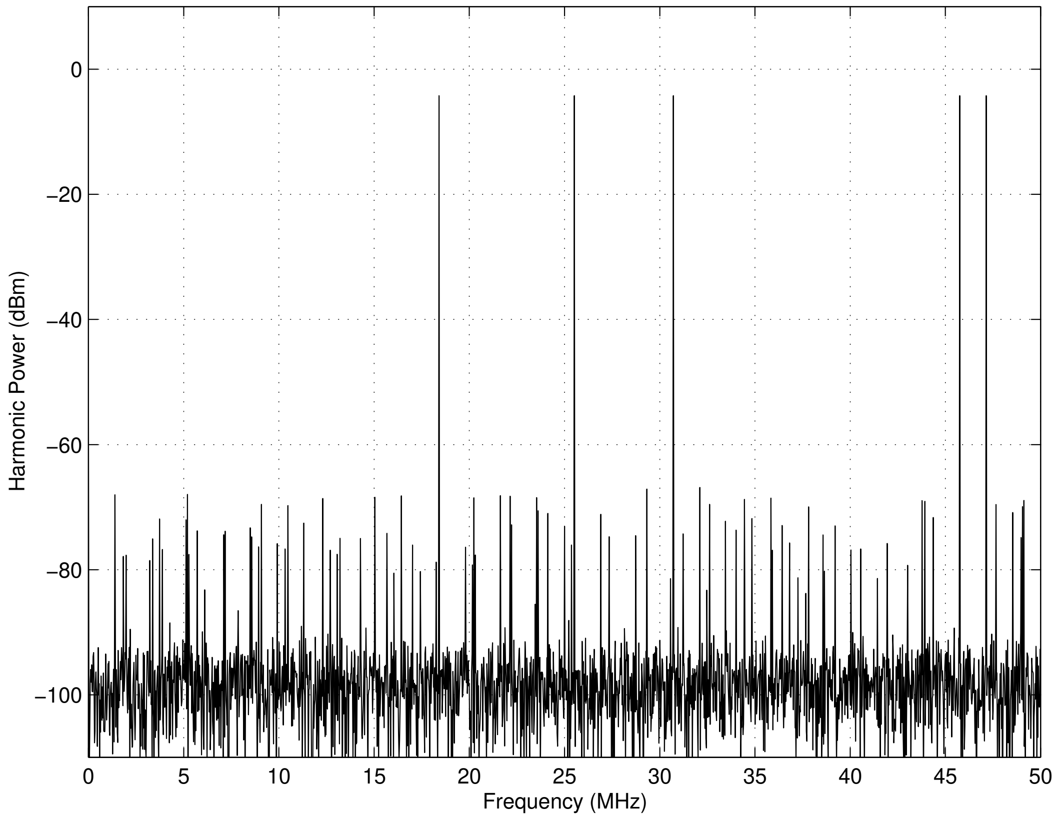

Figure 4.1: Uncompensated simulated DAC output spectrum.

sult shows that this calibration system is capable of modeling and compensating dynamic DAC errors.

4.1.1 Calibration of the DAC

The calibration algorithm described requires that the user set a number of parameters beforehand. These parameters include the number of calibration test signals, the calibration bandwidth, the decimation factor R, and the basis function type. The simulation parameters chosen here were selected to allow a comparison with measured experimental results given in Chapter 5. The decimation factor, R is determined by the constraints of the ADC in the system. R should be chosen such that the ADC sampling is the fastest possible without compromising conversion

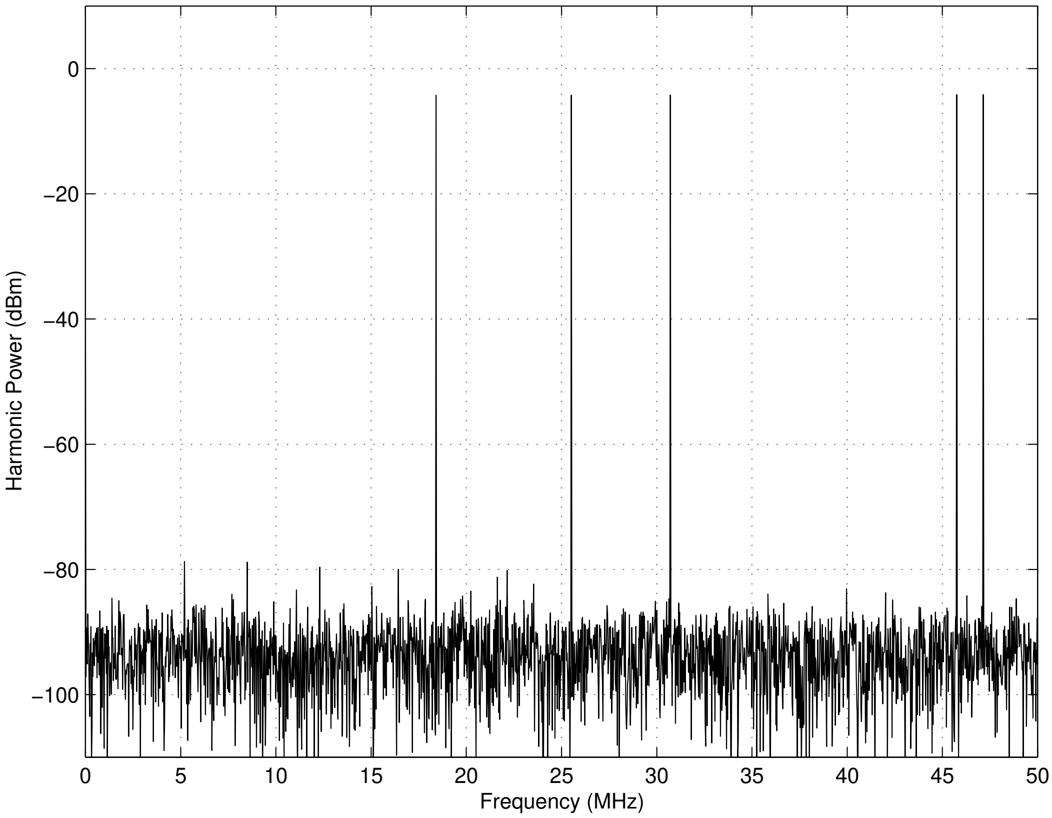

Figure 4.2: Fullband compensated simulated DAC output spectrum.

accuracy. The R for the simulated system is four (allowing a comparison to the R=4 measured results given in Chapter 5).

The calibration bandwidth is determined by the response of the system lowpass filter. This filter should be chosen to cutoff at the Nyquist limit of the ADC to prevent aliasing effects. The simulated filter is a 10th order .004 dB equiripple passband Tchebychev filter with a 6.3 MHz passband width. The simulated DAC output sample rate is 100 MHz, making the ADC sample rate 25 MHz. All frequency bins that lie within the passband of this lowpass filter are usable calibration bins (i.e. are elements of ζ) making the calibration bandwidth the passband of the lowpass filter.

The basis functions used in the simulation are 100 Gaussian basis functions of input and previous input with centers evenly spaced throughout the input space. Equation (4.3) is the basis function output equation for basis function n (b_n) with center (C_x^n, C_y^n) and covariance, σ_n for inputs x_k and x_{k-1}.

$$b_n(x_k, x_{k-1}) = e^{-\frac{(x_k - C_x^n)^2 + (x_{k-1} - C_y^n)^2}{\sigma_n^2}} \tag{4.3}$$

The centers are placed on a 10 by 10 grid spaced evenly throughout the input space. The centers range from zero to 4096 LSBs and are separated by 409.6 LSBs along either the current or previous input axis. The covariance factor, σ_n is equal to 322 for all basis functions in this simulation.

Choosing the number of test signals is simplified by examination of the input space. All calibrations done in this thesis use 100 test signals. The input space is a plot of the previous input vs. current input for all calibration signals. Examination of this plot reveals which combinations of current and previous input the system has used for calibration. Test signals with input and previous input values that are covered in the input space plot are more effectively compensated than those signals

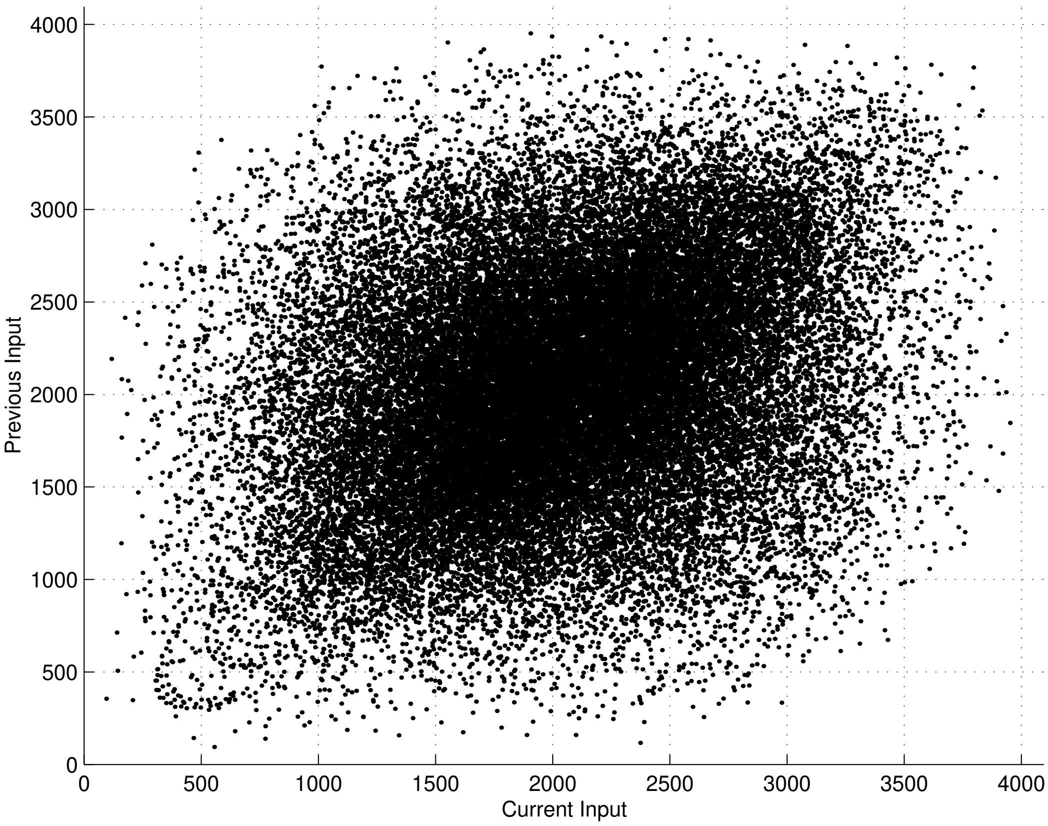

Figure 4.3: Input space for simulated calibration.

with input and previous inputs that land outside of the input space plot. The input space for ten of the 100 input calibration signals that produced the compensated output shown in Figure 4.2 is given in Figure 4.3. Each calibration signal consisted of five-tones selected to produce significant distortion near baseband as described in Section 3.3. The five tones are chosen with one cluster of three calibration tones less than four MHz apart and another cluster of two tones both within 2 MHz of double the lowest frequency tone and within one MHz apart. Let f_1, f_2, f_3, f_4, and f_5 be the five tone frequencies in order from least to greatest. The tones are then related according to equation(4.4).

$$f_3 - f1 \leq 4\text{MHz}$$

$$
\begin{aligned}
f_4 - 2f_1 &\leq 1\text{MHz} \\
f_5 - f_4 &\leq 1\text{MHz}
\end{aligned} \tag{4.4}
$$

The input sequence length and the output sequence lengths are 4096 samples each. This produces a bin mapping factor, k, of four. This input space in Figure 4.3 was constructed with ten of the calibration signals used for the calibration; the additional calibration signals fill in the same area more completely. The number (and to some degree type) of calibration signals selected is based on desired coverage of the input space. Examination of Figure 4.3 reveals that although most of the input space is covered by the calibration signal set, input signals can be chosen that lie outside of the calibration range. An obvious choice is a low frequency single-tone signal that operates near full-scale. This signal has an input space description that closely follows the diagonal (for which $x_k = x_{k-1}$) and stretches close to the corner values where $x_k = x_{k-1} = 0$ and $x_k = x_{k-1} = 2^{D_b} - 1$ where D_b is the number of DAC bits. The input space plot for a single tone signal set on the first frequency bin of a 12-bit DAC super-imposed on the calibration input space plot is shown in Figure 4.4. Compensation of the single tone signal was performed using the same calibration weights used for Figure 4.2. The uncompensated spectrum produced an SFDR of 56.7 dB while the compensated spectrum produced an SFDR of 58.0 dB; which represents a 1.3 dB of SFDR improvement in the compensated signal. This poor performance is the result of an attempt to compensate a signal that has not been accounted for by the range current and previous inputs covered by the calibration signals. Performance for this type of signal tone signal may be improved by choosing calibration signals that cover the same region of the input space.

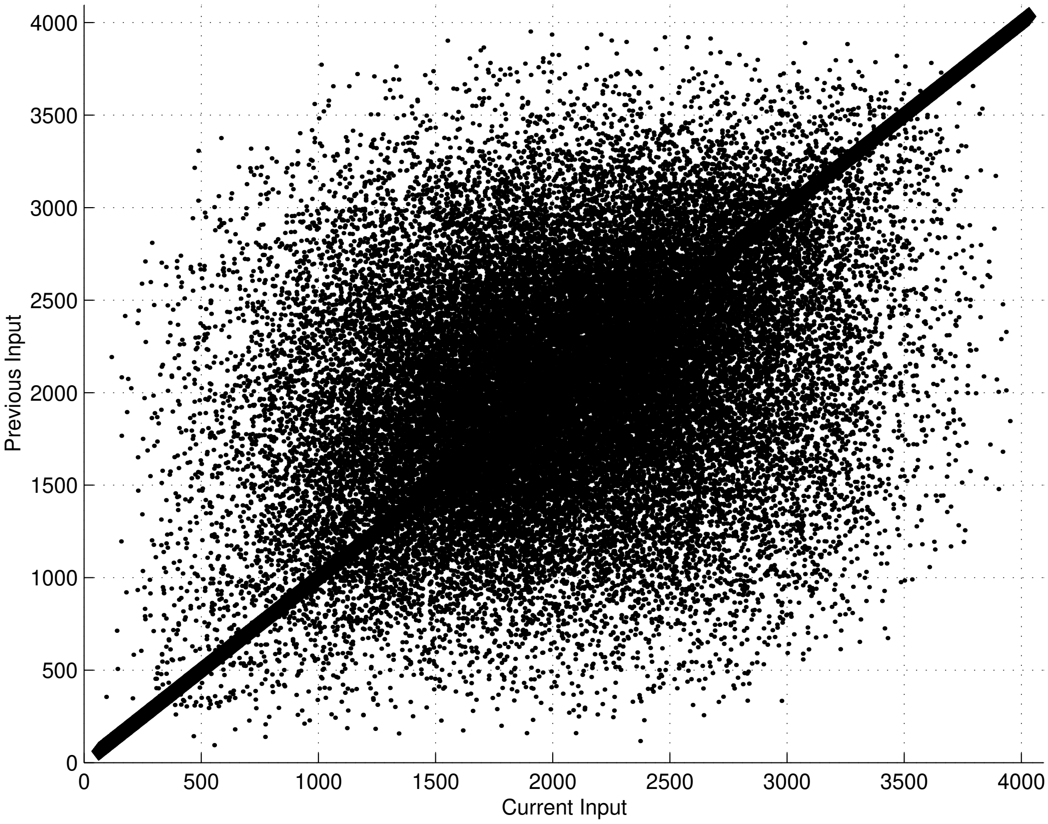

Figure 4.4: Input space for single tone and simulated calibration.

4.2 Ambiguities in the Calibration Solution

The calibration method described thus far has limitations. Particularly disturbing is that the calibration system does not restrict the mean or linear terms of the estimated error characteristic. Since these terms do not create harmonic distortion terms in the calibration regions and no information is provided to the calibration system for the determination of the mean and linear terms, they are arbitrarily large. The arbitrary mean and linear terms may lead to error solutions with compensation values outside the DAC input range. Large mean and linear terms force the compensation algorithm to search longer for the ideal input code. Since the mean and linear terms may be any value it is possible to have a solution that has error values greater than the DAC range, making compensation impossible. An error table (a plot of the error estimate vs. input and previous input) produced by the current algorithm is shown in Figure 4.5. The problems associated with large error table values make it important to restrict the linear and mean terms of the error table. Two possible methods of reducing the effects of this problem are to introduce a solution constraint when solving for the weight vector $\overline{c}$, or to introduce terms in the $\hat{B}_s$ and $\overline{Y}_s$ matrices that discourage large solutions for $\overline{c}$. Implementation of the first method involves adding constraint equations when calculating the pseudo-inverse solution of $\overline{c}$. The second method takes advantage of the fact that calculation of $\overline{c}$ is performing a least-mean-squares solution by adding equations that produce large errors when $\overline{c}$ is large. N_b new equations are added to the system of equations. These equations are each weight times a scale factor, α, equals zero. These equations produce large error values when $\overline{c}$ has large values. To implement these equations a N_b by N_b identity matrix times α is appended to the $\hat{B}_s$ matrix as shown in equation (4.5) and a N_b by one

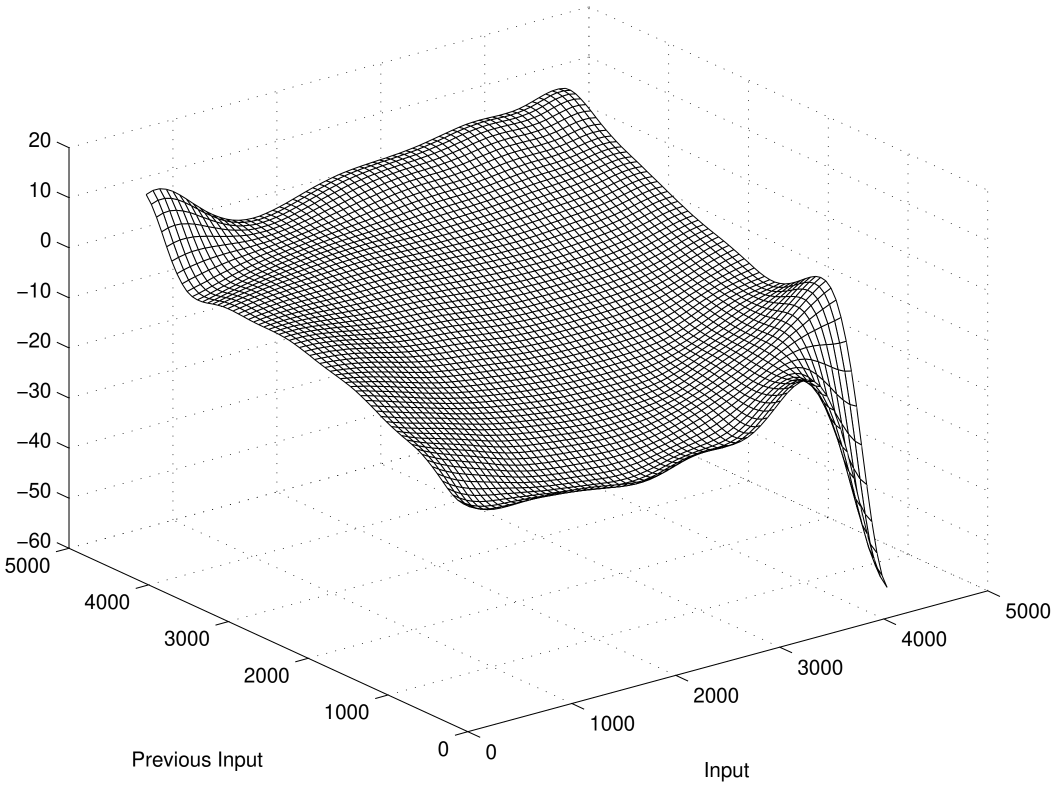

Figure 4.5: Error table for unrestricted weight solution.

vector of zeros is appended to the $\overline{Y}_s$ vector as shown in equation (4.6).

$$\hat{B}_{sm} = \begin{bmatrix} \hat{B}_s \\ \alpha I(N_b) \end{bmatrix}. \tag{4.5}$$

$$\overline{Y}_{sm} = \begin{bmatrix} \overline{Y}_s \\ \overline{0}_{N_b,1} \end{bmatrix}. \tag{4.6}$$

Using the new storage matrices, calculation of $\overline{c}$ is shown in equation (4.7).

$$\overline{c} = (\hat{B}_{sm}^T \hat{B}_{sm})^{-1} \hat{B}_{sm}^T \overline{Y}_{sm} \tag{4.7}$$

The importance of the minimizing equations relative to the regular equations is determined by the scale factor α. When this method is applied to the calibration data of Section 4.1, the error table (Figure 4.6) becomes more reasonable (i.e. has smaller error values) than the non-minimized error table shown in Figure 4.5. Observation of this figure also reveals that the dominant linear component in Figure 4.5 has been removed. This does not compromise performance however, producing an SFDR of 76.0 dB for the same test signal vs. the 74.3 dB SFDR for zero α from Section 4.1. The advantage of the α minimization system is that it performs the desired action more directly; large values of $\overline{c}$ produce large estimated error values resulting in compensation instability, therefore a solution with the minimum $\overline{c}$ values is desired. The disadvantage of the α minimization to the constraint method is that α minimization introduces another unknown parameter, α into the calibration. The advantage of α minimization outweighs the disadvantage by eliminating the arbitrary selection of the mean and linear error terms, the compensation algorithm becomes stable. Now, in order to generalize the algorithm it is important to examine the effects a reduced calibration bandwidth (as set by the lowpass filter) has on calibration performance.

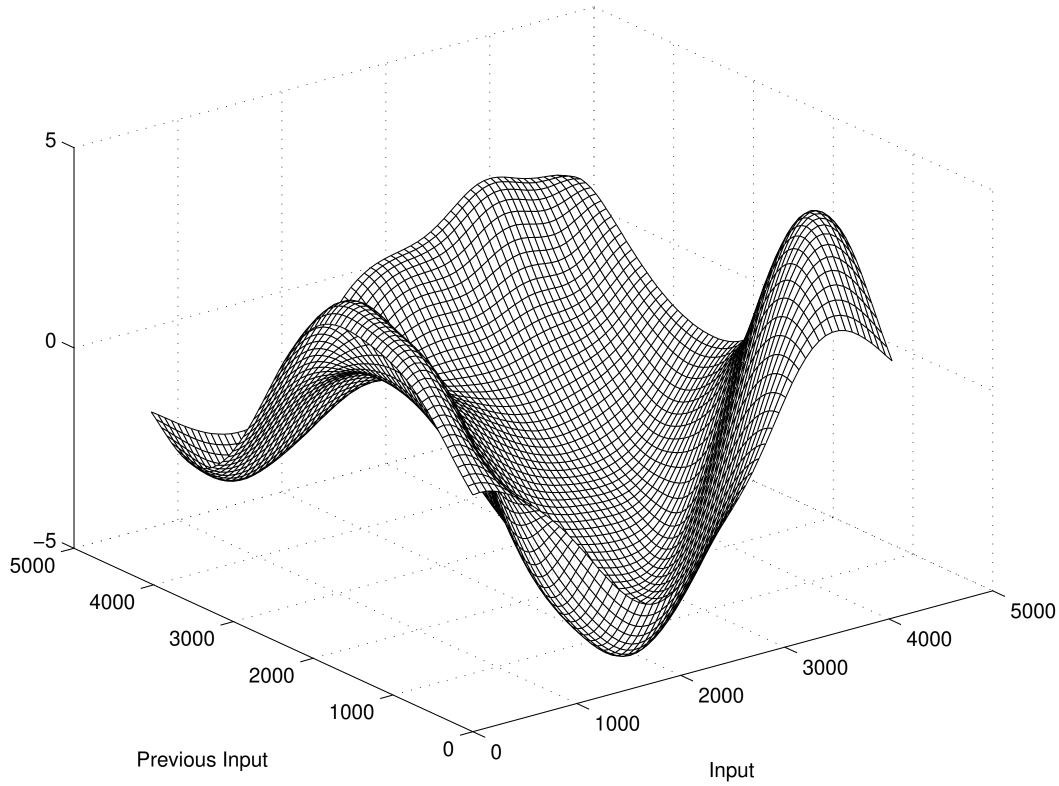

Figure 4.6: Error table for α restricted weights, $\alpha = 50$.

4.3 Effect of Measurement Bandwidth

For calibration bandwidths that are a significant percentage (greater than 2%) of the DAC sampling rate, the current calibration algorithm is effective. When the calibration bandwidth is chosen to be small (less than 2% of DAC sampling rate) calibration performance becomes hampered by the effects of the limited amount of information available to the calibration. For highly limited bandwidth, the solution for the weights that produce the same baseband response is no longer unique. The estimated error characteristic may produce a lowpass filtered response that provides adequate estimation of error terms in the calibration band, but also introduces extraneous error terms that are significant only outside the calibration region. To describe how this happens, let the converter error be a purely static function, $S(x_k)$. Now assume that the estimated error ($\hat{E}(x_k, x_{k-1})$) solves for a lowpass solution of $S(x_k)$ shown in equation (4.8).

$$\hat{E}(x_k, x_{k-1}) = \frac{1}{2}S(x_k) + \frac{1}{2}S(x_{k-1}) + H(x_k) - H(x_k - 1) \tag{4.8}$$

In equation (4.8) $H(x_k)$ is an arbitrary function created by the calibration algorithm. The solution in equation (4.8) produces the same baseband error terms as $S(x_k)$, but as the difference between x_k and x_{k-1} becomes large, high frequency error terms appear that are not present inside the calibration band. To observe the system producing this type of result the converter is given a static error only, shown in equation (4.9).

$$S(x_k) = q(\beta + \gamma x_k)^2 \tag{4.9}$$

The basis functions used to estimate this error are ten Gaussian functions of the current input code, and ten Gaussian functions of the previous input code (all basis function centers are evenly spaced over all possible input code values). A

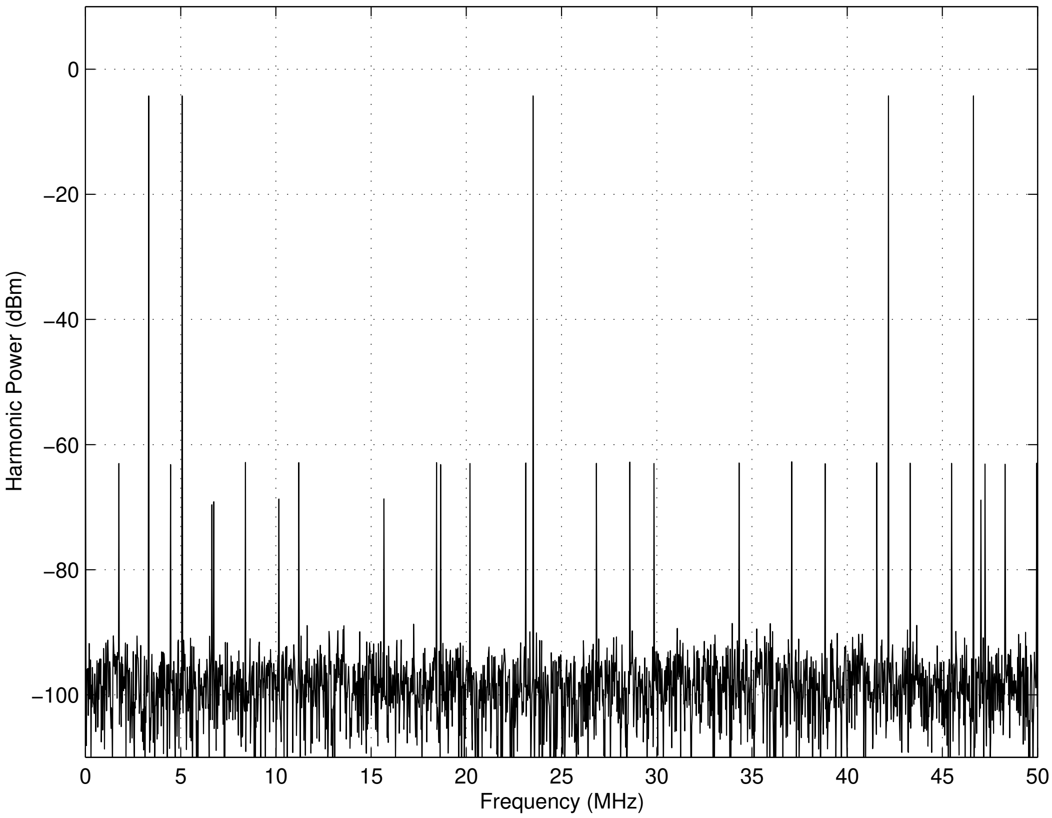

Figure 4.7: Uncompensated spectrum for small bandwidth calibration.

compensated frequency response of this DAC, for $q = .006$ and calibration bandwidth set to 1% of the DAC sampling rate, is shown in Figure 4.8 along with the uncompensated response shown in Figure 4.7. Notice that within the calibration range (1MHz) the error terms are below the noise floor of -90 dBm, but as frequency increases, error is introduced by the compensation. A plot of the actual error function, $S(x_k)$, vs. input code is shown in Figure 4.9. This plot also shows the estimate of error vs. current input, the estimate of error vs. previous input, and the sum of the two. Notice that the sum of the two errors models the actual error curve (with mean and linear differences that do not affect performance). The solution shown in Figure 4.9 makes the error closely model the correct result for DC signals (where previous input equals current input) but stray from the correct

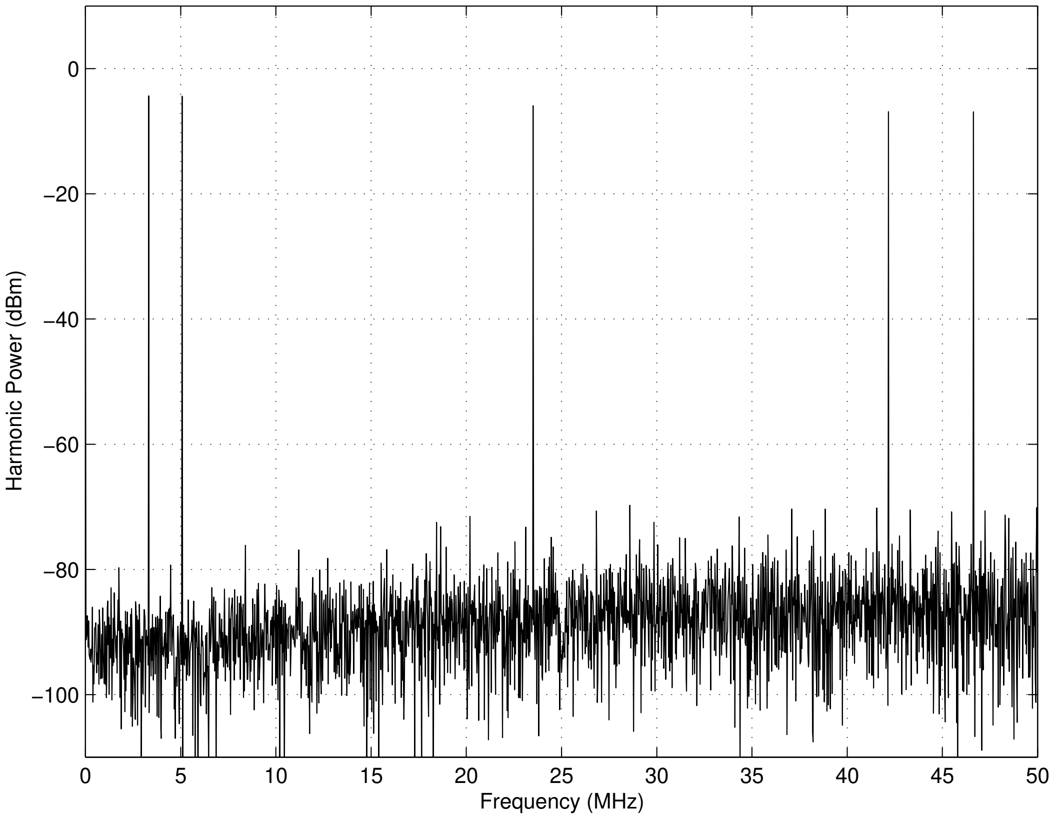

Figure 4.8: Compensated spectrum for small bandwidth calibration.

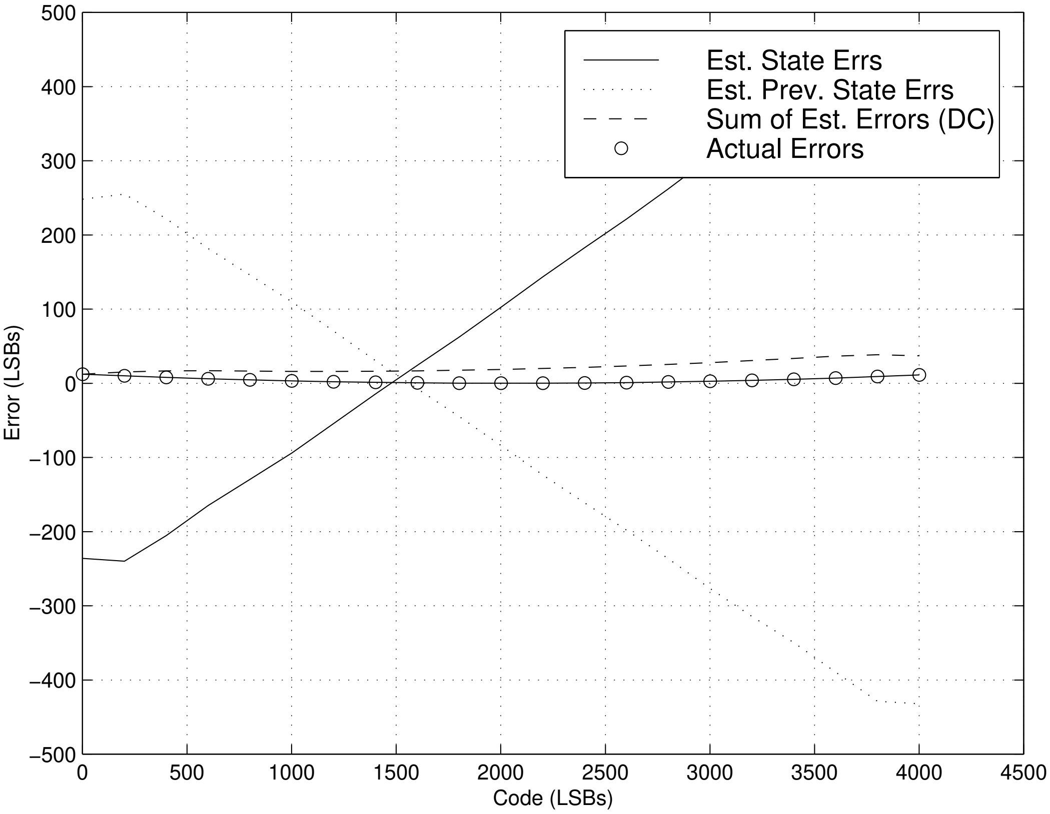

Figure 4.9: True error curve and estimated errors vs. code.

solution as the difference between x_k and x_{k-1} becomes large.

Performance of the current system, for highly limited bandwidth, is unacceptable because it creates error terms that can dominate converter errors. To prevent this from happening two corrections must be added to the calibration algorithm. To prevent the arbitrary error function form $(H(x_k) - H(x_{k-1}))$ from occurring, the minimum weight values that provide an error solution are chosen. A method for finding minimum weight values is the α minimization technique described in Section 4.2. The minimum weight solution eliminates the extraneous function $H(x_k) - H(x_{k-1})$ that harms compensation performance outside the calibration band. A second step to preventing an erroneous solution is to perform a two-stage calibration. This two-stage calibration first eliminates errors that are a function of the current input only, then performs the dynamic calibration on the statically compensated signal. Implementing the α minimization and the two-stage calibration system produces a compensated response shown in Figure 4.10 for the same DAC previously modeled; that is, with $S(x_k)$ error only, and a 1bandwidth. Observation of the two-stage with α minimization compensated DAC output (Figure 4.10) reveals that it is more effective than the single stage without α minimization shown in Figure 4.8. Static DAC errors are easier to measure than dynamic errors making the current calibration algorithm inefficient for measuring the static error for the first stage static calibration. The following section describes a method of measuring static DAC errors.

4.3.1 Obtaining a Static DAC Characteristic

Measuring static DAC error is easier than measuring the dynamic error because it can be measured using very low frequency signals within the calibration band of the system. Under the assumption that the static error characteristic is practically equivalent to the error characteristic for a single tone signal on the first

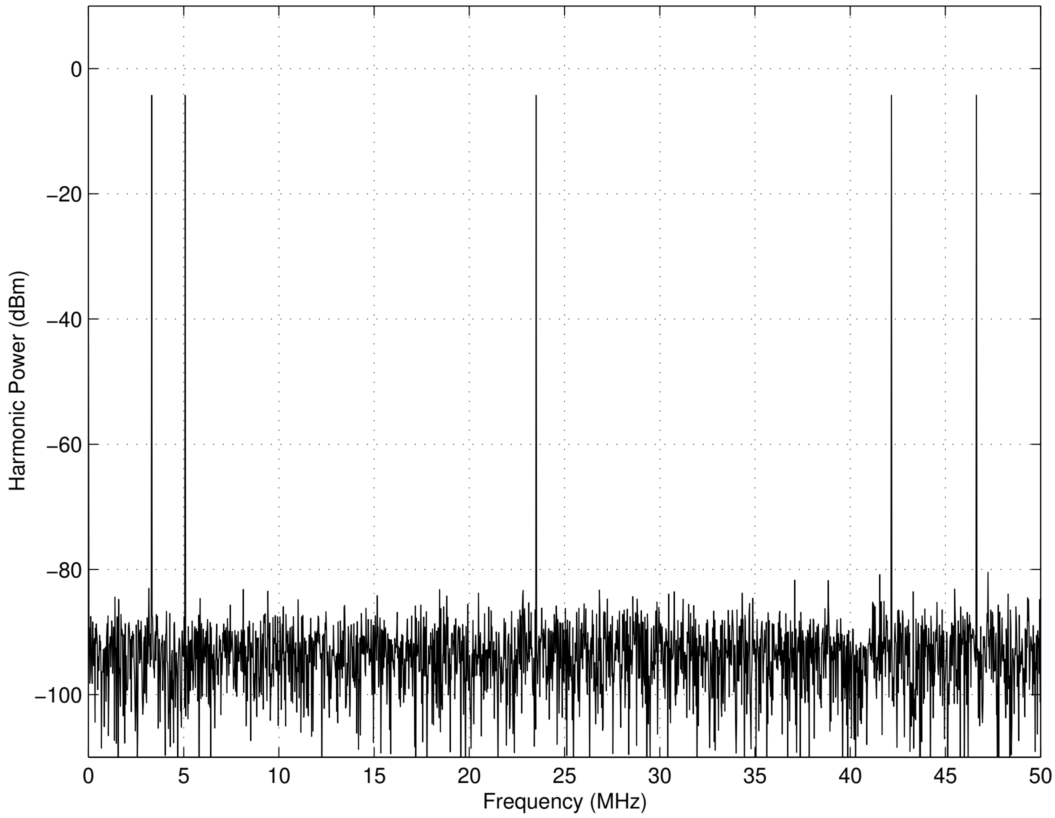

Figure 4.10: Two-stage compensated spectrum for small bandwidth calibration.

frequency bin, the static error may be measured using a single tone test. First the single tone on the first frequency bin is passed to the input of the DAC. The ADC output for this single tone is retrieved and a DFT is taken. The values for the first 20 harmonics, excluding the fundamental are kept and the rest are zeroed. A frequency domain error vector is initialized to a N_{sD} long zero vector. These 20 harmonic values are divided by their respective linear characterization values and stored in bins 2 through 21 of the frequency domain vector and their conjugates are stored in the corresponding conjugate bin numbers. The error waveform is the inverse DFT of the resulting frequency domain vector. The ith element of this waveform vector is the error associated with input x_i. For each input value to the DAC the error values at each time the DAC input hit that value is averaged. This produces the error for each input value the single tone input sequence hits. Error values for inputs the single tone input sequence does not contain are found by linearly interpolating between the nearest calibrated inputs. Let I_1 be the next smallest calibrated input value from the desired input value I_n. Also let I_2 be the next greatest calibrated input value from the desired input value I_n. In equation (4.10) $\hat{E}_{st}(I)$ is the static error associated with input I.

$$\hat{E}_{st}(I_n) = \frac{\hat{E}_{st}(I_2) - \hat{E}_{st}(I_1)}{I_2 - I_1}(I_n - I_1) + \hat{E}_{st}(I_1) \tag{4.10}$$

Values outside the range of the input signal (greater than the highest input or lower than the lowest input) are set to the last value inside the signal range. This static error sequence can now be used to provide statically compensated samples (using techniques described in Section 2.4) to the dynamic calibration algorithm for use in the two-stage calibration described in Section 4.3.

4.4 A Refined Calibration Algorithm

The limitations of the calibration algorithm described in Section 3.2 reveal a need to improve the calibration algorithm using techniques described in Sections 4.2 and 4.3. The α minimization must be added to the calibration to regulate the mean and linear components of the error table, as well as prevent the introduction of arbitrary functions when the calibration bandwidth is highly limited. Additionally, when the calibration bandwidth is limited to less than 2% of the DAC output sampling rate a two stage calibration can be used to avoid a lowpass error solution that degrades with increasing frequency. The two stage calibration compensates the static component of the error then performs the dynamic calibration on the remaining error. A step by step refined calibration process is given.

1. Select a calibration signal set that covers the input space and produces low-frequency intermodulation terms.

2. Obtain DAC static error by performing a static calibration as described in Section 4.3.1.

3. Generate a multi-tone test signal, $\overline{x}$. This signal is structured as described in Section 3.3 to produce low-frequency intermodulation error components.

4. Statically compensate using the static compensation procedure described in Section 4.3.1 and the compensation procedure described in Section 2.4.

5. Apply this statically compensated sequence to the DAC input of the DAC calibration system described in Section 3.1.

6. Collect a buffer of data from the ADC, $\overline{y}$.

7. Calculate the basis function output matrix, $\tilde{B}$, as described in Section 3.2.4, for the statically compensated input sequence.

8. Select the calibration bins, b_i. Take a DFT for the calibration bins of $\tilde{B}$ and $\overline{y}$ to obtain $\tilde{G}$ and $\overline{Y}$ as described in Section 3.2.4, equation (3.19) and equation (3.21).

9. Multiply the basis function DFT output matrix, $\tilde{G}$ by the linear characterization matrix for the selected bins, $\tilde{L}_{\overline{b}_i}$, to obtain $\hat{B}$ as described in Section 3.2.4, equation (3.22).

10. Store the real and imaginary parts of $\hat{B}$ and $\overline{Y}$ separately in the storage variables $\hat{B}_s$ and $\overline{Y}_s$ as discussed in Section 3.2.4, equations (3.23) and (3.24).

11. Repeat steps 3 through 10 until all test signals have been applied to the DAC.

12. Append α minimization equations to $\hat{B}_s$ and $\overline{Y}_s$ as discussed in Section 4.2, equations (4.5) and (4.6), to produce $\hat{B}_{sm}$ and $\overline{Y}_{sm}$.

13. Calculate the basis function weights, $\overline{c}$ using a pseudo-inverse of the $\hat{B}_{sm}$ matrix shown in equation (4.7).

14. Compute the estimated error using equation (3.12).

15. Compensate any input sequence with this estimate of the error and the techniques described in Section 2.4.

CHAPTER 5

Measured Results

The simulation results show that dynamic error functions of input and previous input that create distortion terms near baseband can be compensated with the calibration scheme described in Chapter 3 with improvements discussed in Chapter 4. Now that the calibration has successfully compensated a simulated DAC, the next step is to explore calibration and compensation of a real DAC. This chapter will talk about the hardware implementation used for this thesis. Static error compensation results will be presented followed by dynamic compensation results. Finally the effects of α selection on compensation performance will be investigated.

5.1 Description of a Hardware Implementation

The lowpass implementation described in Section 3.1 is the model used for the hardware system. The device under test for the measured results presented is a Tektronics TKDA30P 12-bit DAC with clocked output operating at 100 MHz. A Hewlett Packard HP16522A pattern generator is used to supply input codes to the DAC. An wideband matched input impedance elliptic lowpass filter with cutoff frequency 6.3 MHz and minimum stopband frequency 8 MHz is used for the calibration range filter. This filter has a circuit added to it to make the input resistance nearly 50 Ohms over all frequencies. This is required to prevent large signal reflections for high frequency tones the filter removes. The ADC is an Analog Devices AD9042D 12-bit ADC with a sampling clock at 25 MHz, a quarter of the DAC output clock rate. ADC output data is retrieved from a Tektronics 9504 data cache. Calibration of the DAC was performed using 100 five-tone test signals. The five tone signals are chosen with one cluster of three test signals less than 4 MHz apart and another cluster of two test signals within 2 MHz of double the lowest frequency tone and within 1 MHz apart as described in equation (4.4). The DAC input sequence and the ADC output sequence are both 4096 samples long. The bin mapping factor, k, for this system is, according to equation (3.5), equal to four. α minimization is used to obtain the minimum weight solution. Unless otherwise specified α is 50. In order to make the phase response meaningful a method of obtaining a consistent phase reference is necessary when implementing the hardware.

5.1.1 Obtaining a Consistent Phase Reference

Before the system phase response may be measured a method for maintaining a consistent phase reference must be available. The implementation used for this hardware system is the addition of timing bits produced by the pattern generator at the DAC input that are captured by the data cache which captures the ADC output. Since the ADC bits are captured at a slower rate than the DAC input sampling rate (and hence the pattern generator output rate) a method of determining a fractional delay at the ADC using the timing data is necessary. The timing bits must have enough resolution to cover the decimation factor, R. For example if $R = 4$ then two timing bits are required; if $R = 8$ then three bits are needed. The timing bits count from zero to $R - 1$ (in binary format) then are all zeros until the last R input samples. The last R samples have the least significant bit set to 1. The timing bits are captured by the data cache at the same time as the ADC samples. The data cache output for the timing bits is a series of zeros with a one and number between zero and $R - 1$ adjacent to it for every period of the DAC output sequence. The location of the one provides the user with an integer number of delays for the output sequence, the second number provides the amount of fractional delay to remove (a fractional delay is necessary to make the sequence line up with the DAC input samples changing at a faster clock rate). Let q be the sample number where the first one is found in the timing sequence. Let l be the value of the following timing data sample. A consistent time reference may be obtained by removing the integer and fractional delays from the output samples. Let $\overline{u}$ be the vector of samples retrieved from the cache. The output samples with a consistent time reference, $\overline{y}$, may be obtained by first taking the DFT of the cache output, producing $\overline{U}$. These samples are multiplied by their respective delay factors to produce $\overline{Y}$. For the positive frequency bins the value of Y_k for bin k is given by equation (5.1).

$$Y_k = e^{2\pi jk\frac{q-\frac{l}{R}}{N_sA}}U_k \tag{5.1}$$

The negative frequency bins are simply multiplied by the conjugate terms of the corresponding positive frequency domain delay coefficients. $\overline{y}$ is then calculated by taking an inverse DFT of $\overline{Y}$. With a consistent time reference it is possible to obtain a reasonable measure of the system phase response.

5.2 Static Compensation Results

The first compensation discussed involves compensating the static component of the DAC error. This static error is measured using the static calibration system described in Section 4.3.1. The measured errors are compensated using the method described in Section 2.4. Figure (5.1) shows the DAC output spectrum for five uncompensated test tones placed on random bins. After static compensation the output spectrum in Figure 5.2 is obtained. To ease comparison between the uncompensated and the compensated spectral performance we generate spur reduction plot. This plot is the result of taking all harmonics above the noise floor (set to -77 dBm for this examples) and subtracting the compensated response (in

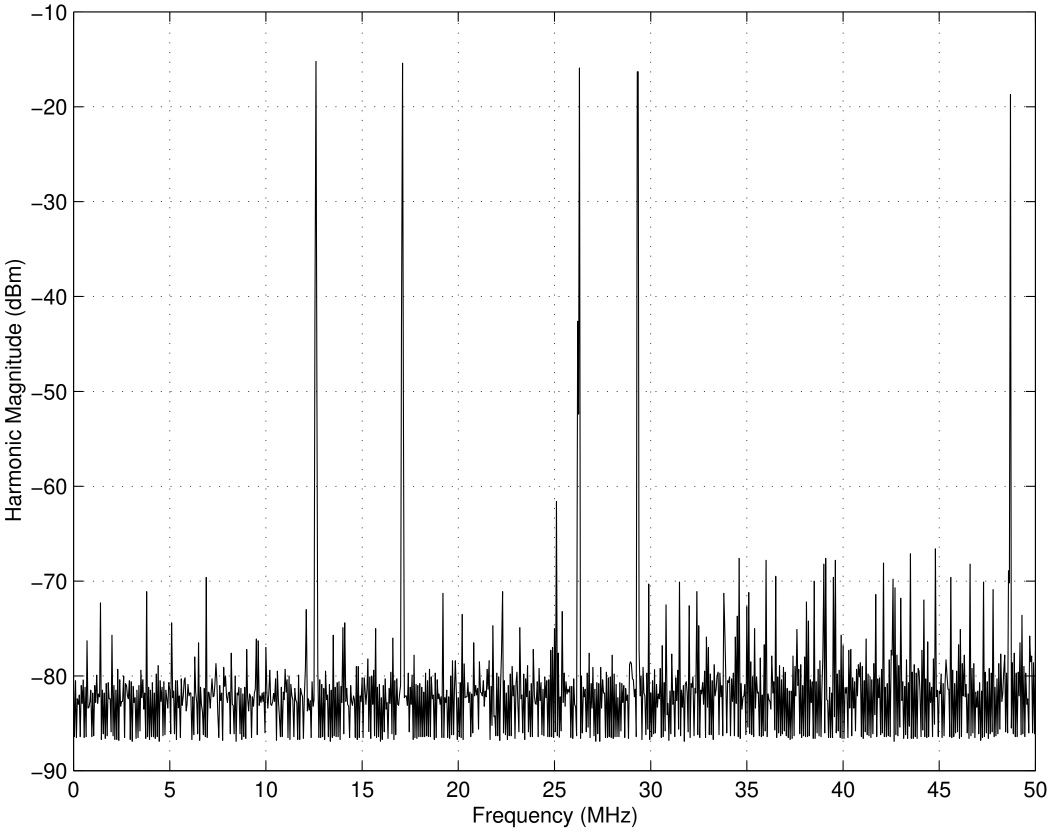

Figure 5.1: Uncompensated DAC output spectrum.

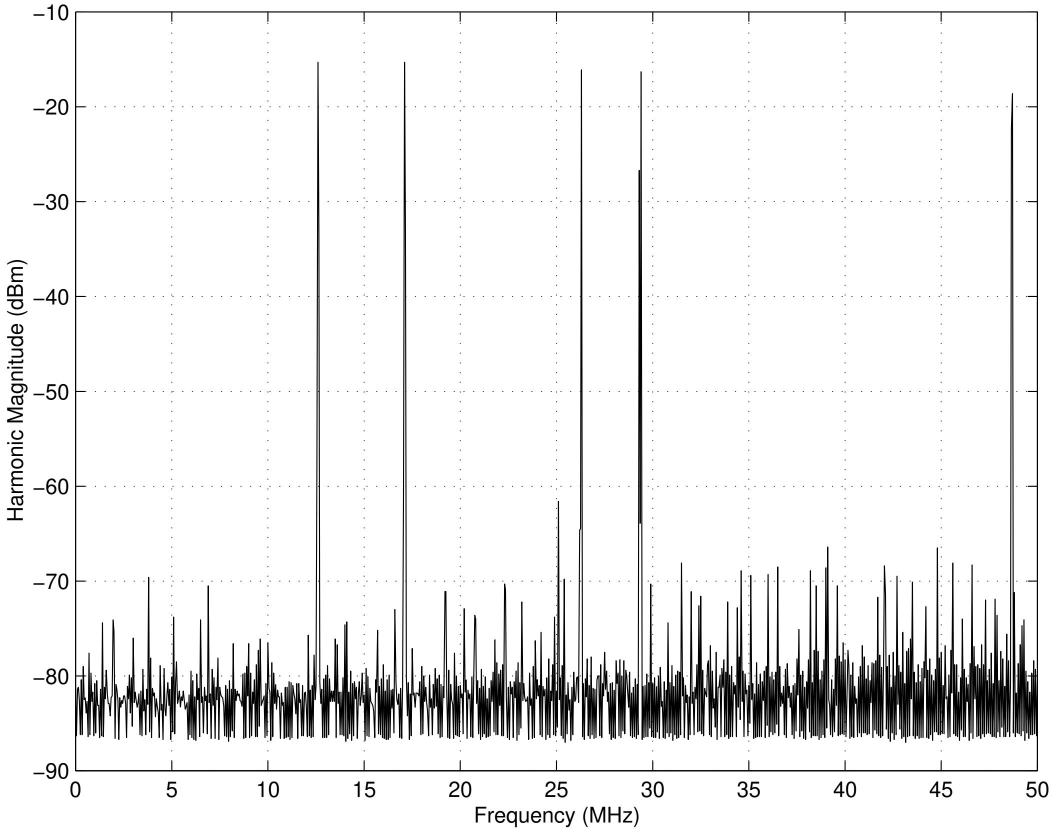

Figure 5.2: Statically compensated DAC output spectrum.

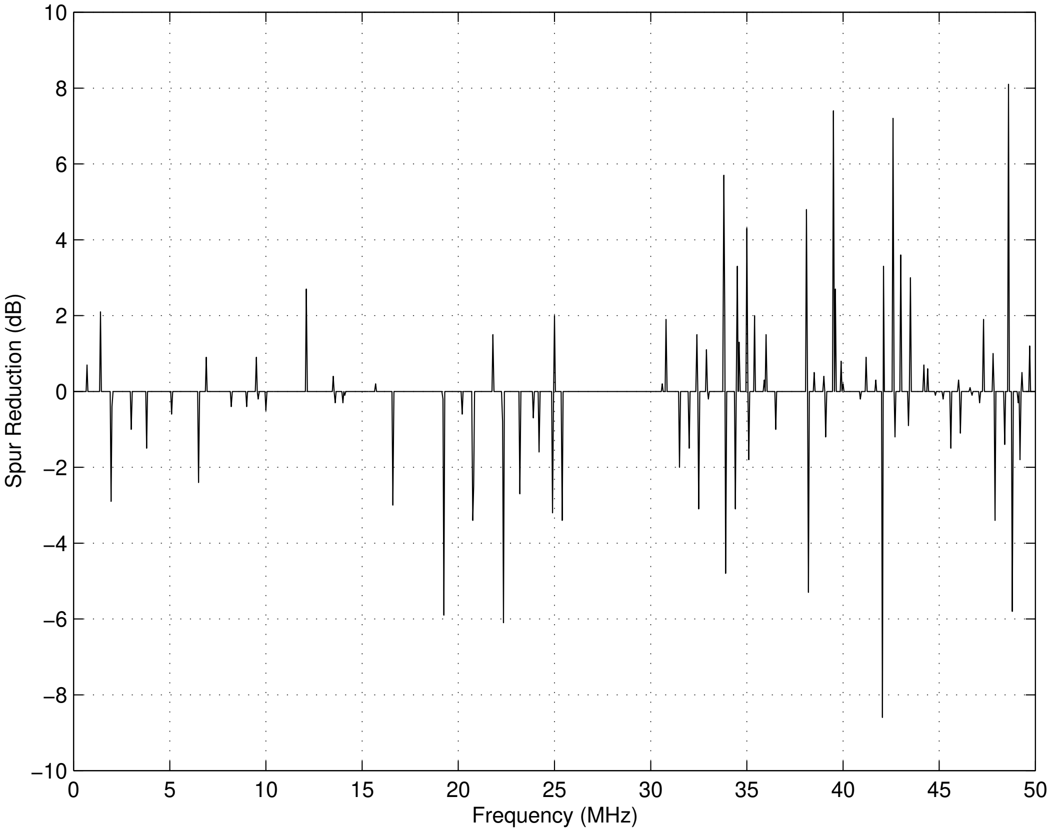

Figure 5.3: Spur reduction for static calibration.

dBm) from the uncompensated response (in dBm). Positive values in this plot are spurs reduced by the compensation, while negative values indicate the compensation has worsened those frequency terms. The spur reduction plot for the static calibration is shown in Figure 5.3. It is apparent from this result that a static error model is not sufficient for modeling the converter error. There is no noticeable spectral improvement in the compensated spectrum. The next step is to calibrate using dynamic error calibration methods.

5.3 Dynamic Compensation Performance

The dynamic calibration solves for the error as a function of current and previous inputs to the converter as described in Section 4.4. Calibrating the DAC

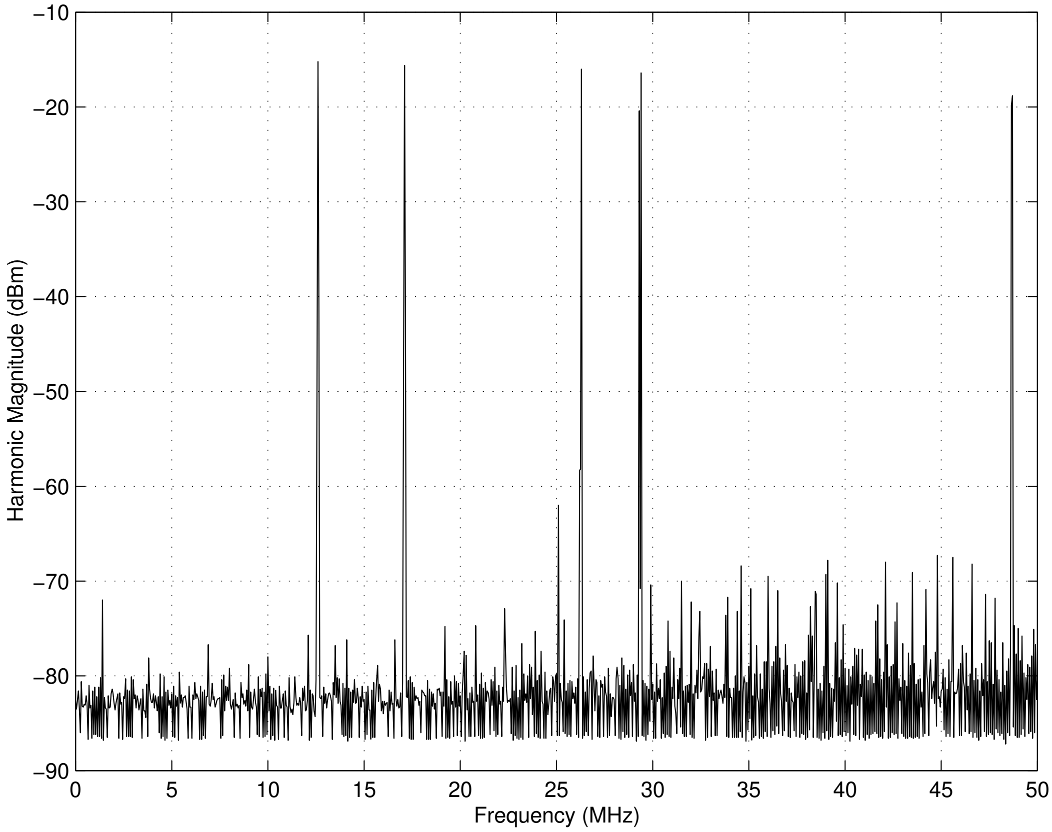

Figure 5.4: Dynamically compensated DAC output spectrum.

using these methods and compensating it using the dynamic error model produces the output spectrum shown in Figure 5.4. Although the dynamic compensation does not remove all harmonic distortion it does remove much distortion in the 6.3 MHz calibration band. This is evident in the spur reduction plot shown in Figure 5.5. Obviously there are distortion terms that are not compensated by the algorithm. The next section discusses some possible reasons these distortions are not compensated.

5.3.1 Undetectable Error Mechanisms

From observation of Figure 5.4 it is apparent that there are error mechanisms not compensated by this dynamic compensation scheme. There are two

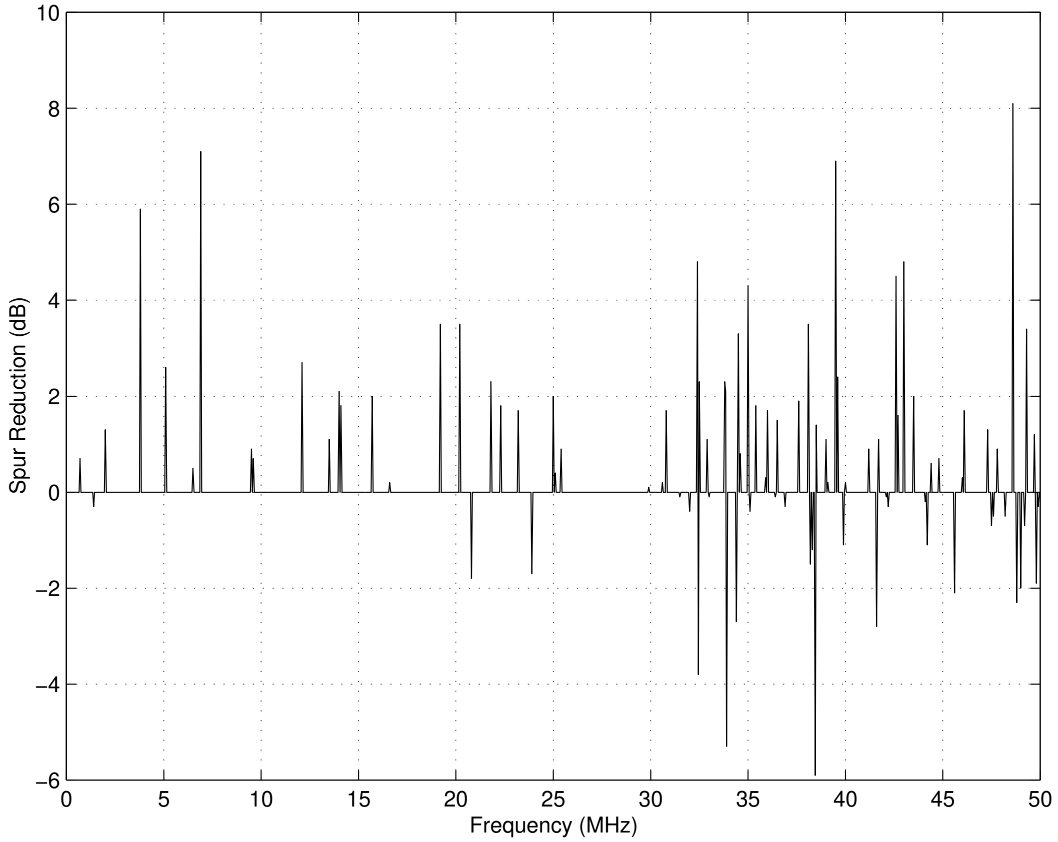

Figure 5.5: Spur reduction for two-stage dynamic calibration.

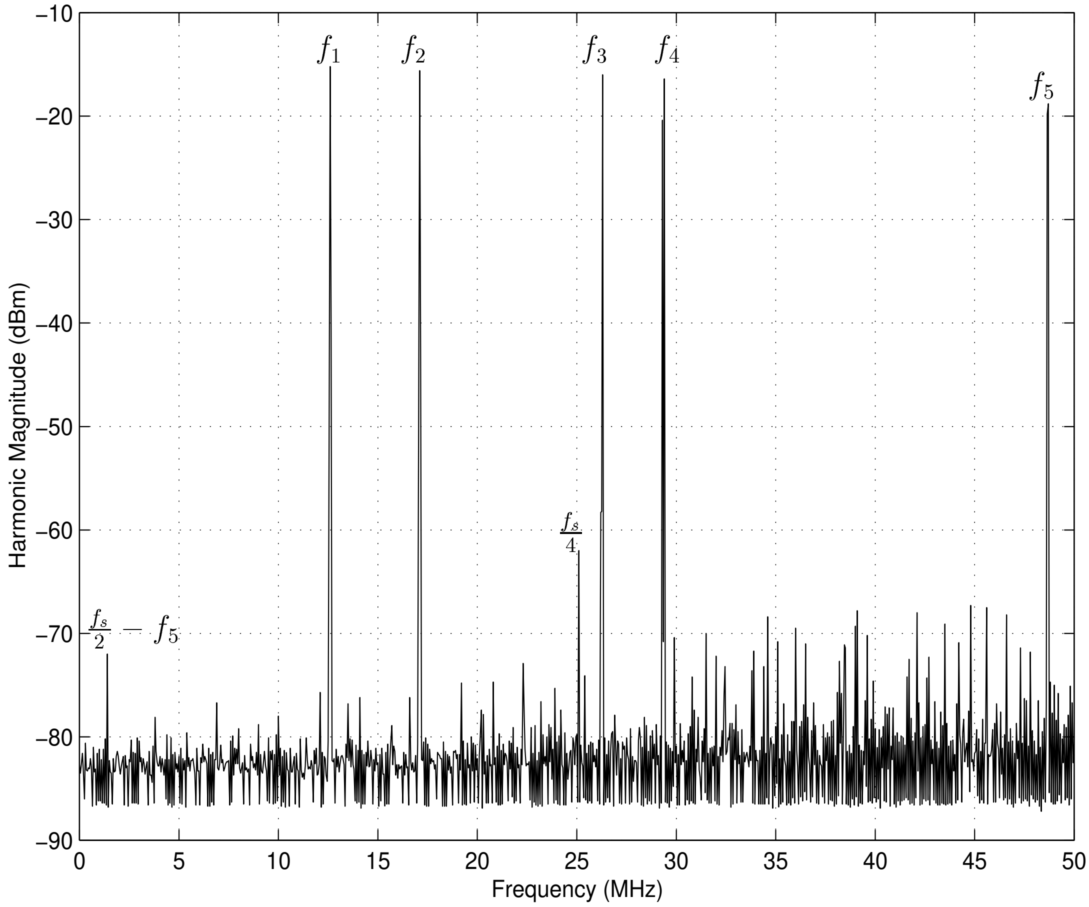

Figure 5.6: Identification of some undetectable error mechanisms.

possible reasons that the errors are not compensated. First the error may not be a function of input and previous input. Some examples of this type of error are the intermodulation with the clock signal, clock signal bleed-through, clock jitter, and power supply variations. The compensated output of this converter displays at least the intermodulation with clock signals error as well as some clock bleed-through, as shown by the terms labeled $\frac{f_s}{4}$ and $\frac{f_s}{2} - f_5$ in Figure 5.6. The other possibility is that error may be a function of current input and previous input to the converter, but error terms near baseband are always below the noise floor within the calibration band. An example of this effect is a high pass filtered static characteristic of the form shown in equation (5.2).

$$E(x_k, x_{k-1}) = H(x_k) - H(x_{k-1}) \tag{5.2}$$

This error form is possible in real converters; the bit transition effect model discussed in Section 2.3.2 is one possible error model with this form. This error is a function of the input and previous input, but will not produce error above the noise floor within the calibration range unless the calibration range contains some higher frequency terms where these errors are above the noise floor. The converter compensated may have some of these out-of-band errors that produce some of the harmonic distortion seen outside the calibration band in Figure 5.6.

5.4 Selection of α

One design choice to make when using this calibration and compensation scheme is the selection of the α minimization factor. The goal when choosing an α value is to obtain a well-behaved solution to simplify compensation while not compromising performance. When α is too small the solution for the error may have large arbitrary mean and linear terms, while an α set too large will overly limit the solution and performance will be compromised. As α increases to infinity the solution approaches the uncompensated result, i.e. the error table approaches zeros for all values of input and previous input. To gain an understanding of the algorithm's sensitivity to the minimization factor, several α values are tested. Figure 5.8 shows the output spectrum for $\alpha = 0$ compared to the uncompensated response in Figure 5.7. This result provides maximum compensation performance at the expense of instability in the algorithm. To gain insight about this improvement a spur reduction plot is generated, shown in Figure 5.9. In this case all error values fall within the converter range. This is not always true for $\alpha = 0$; a plot of

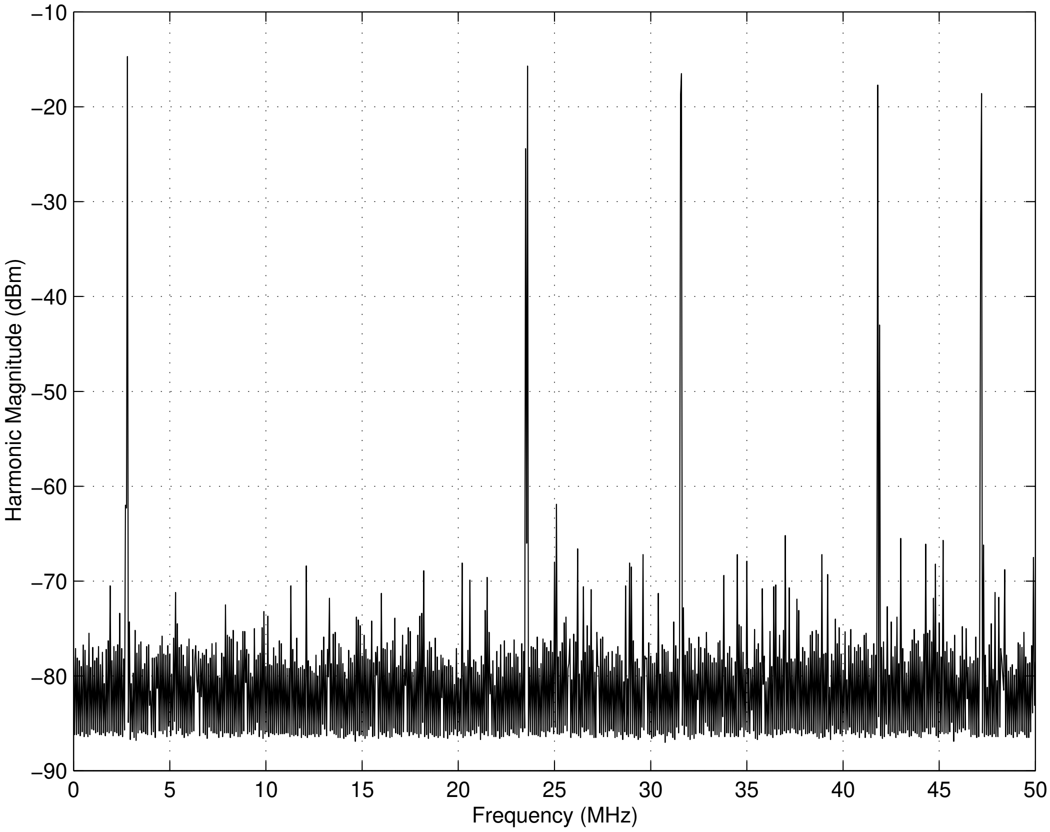

Figure 5.7: Uncompensated DAC output spectrum.

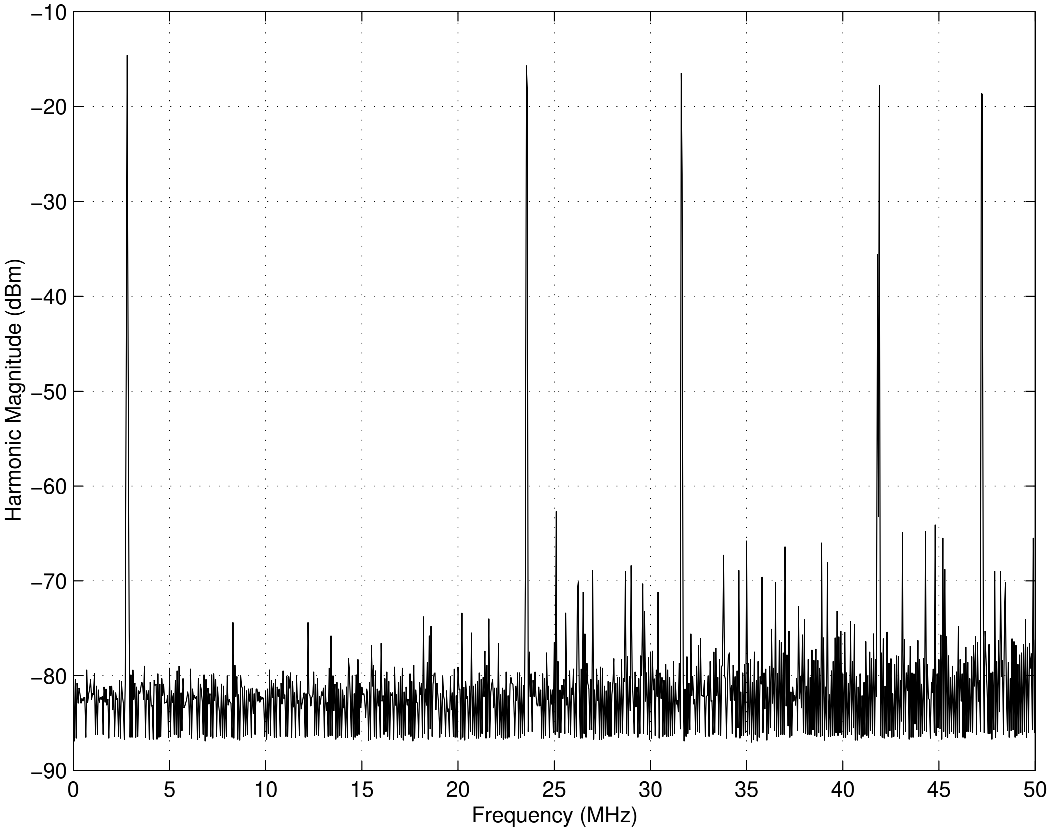

Figure 5.8: Compensated DAC output spectrum, $\alpha = 0$.

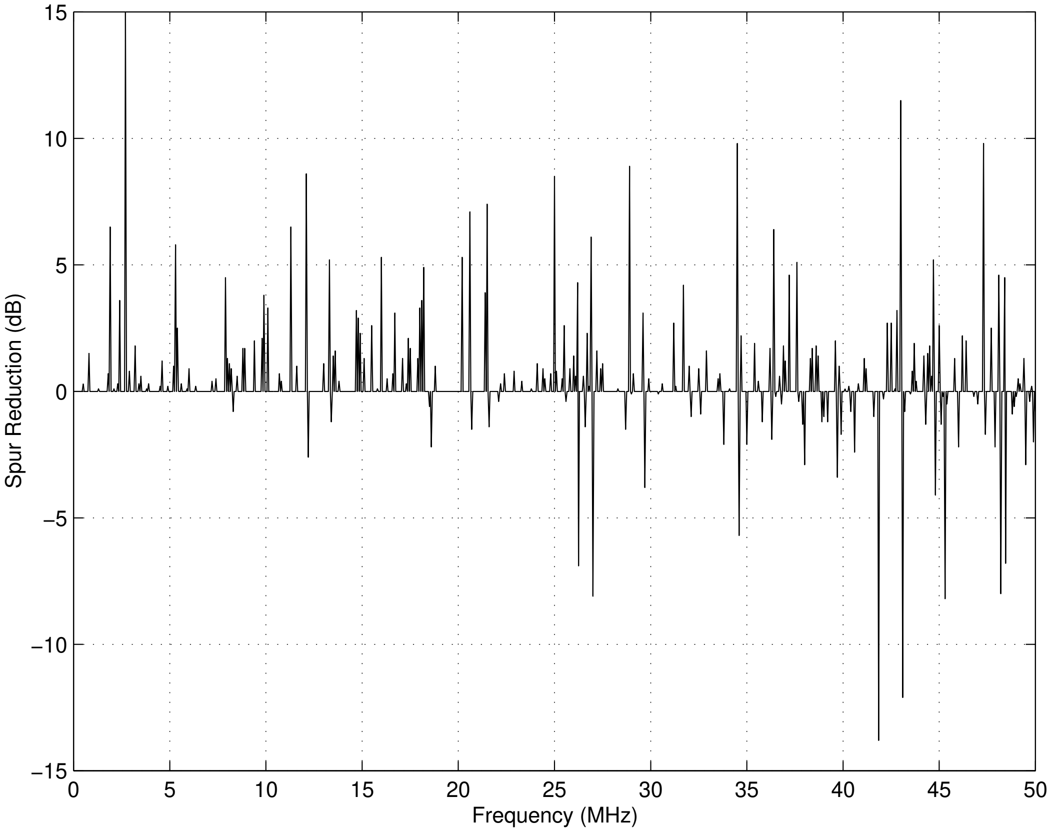

Figure 5.9: Spur reduction for zero α.

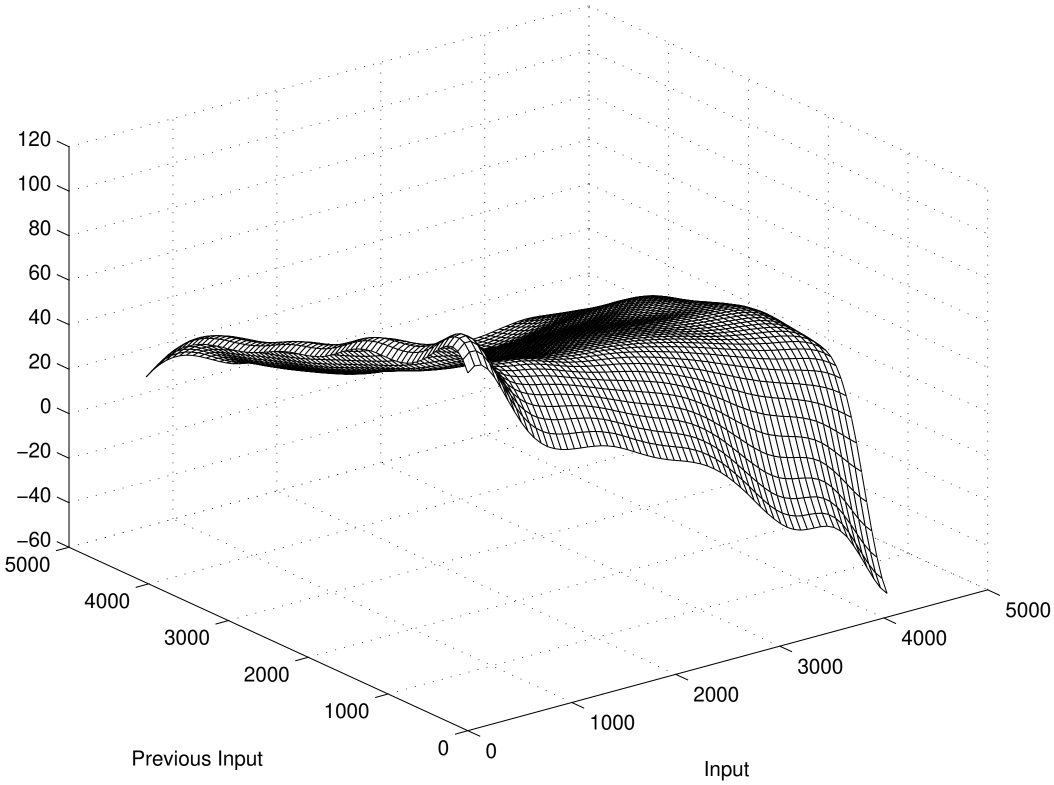

Figure 5.10: Error table for $\alpha = 0$.

the error table for $\alpha = 0$ is shown in Figure 5.10. The mean and linear component in the error table may be arbitrarily chosen so large that compensation is not possible. Additionally, for a more limited bandwidth, compensation performance can be compromised by arbitrary functions introduced into the solution as discussed in Section 4.3. With this in mind, α is increased to 50 for Figure 5.11. An α of 50 produces practically equivalent performance as an α of zero without arbitrary factors in the error table. The spur reduction plot for $\alpha = 50$ is shown in Figure 5.12. This plot shows similar performance to the $\alpha = 0$ case shown in Figure 5.9. The error table for α equal to 50 (Figure 5.13) has smaller values than the zero case (Figure 5.10). This makes the compensation more stable (i.e. has no large arbitrary components) and prevents the introduction of arbitrary functions in the solution. A calibration for $\alpha = 400$ provides the spectrum shown in Figure 5.14.

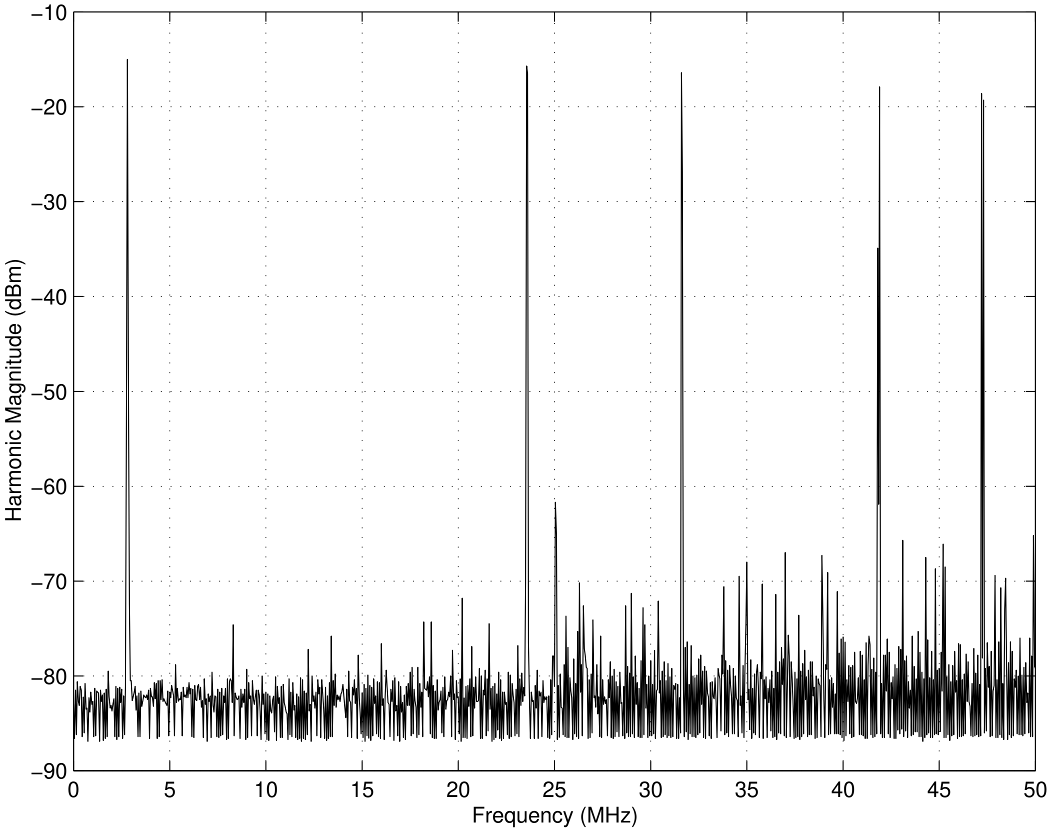

Figure 5.11: Compensated DAC output spectrum, $\alpha = 50$.

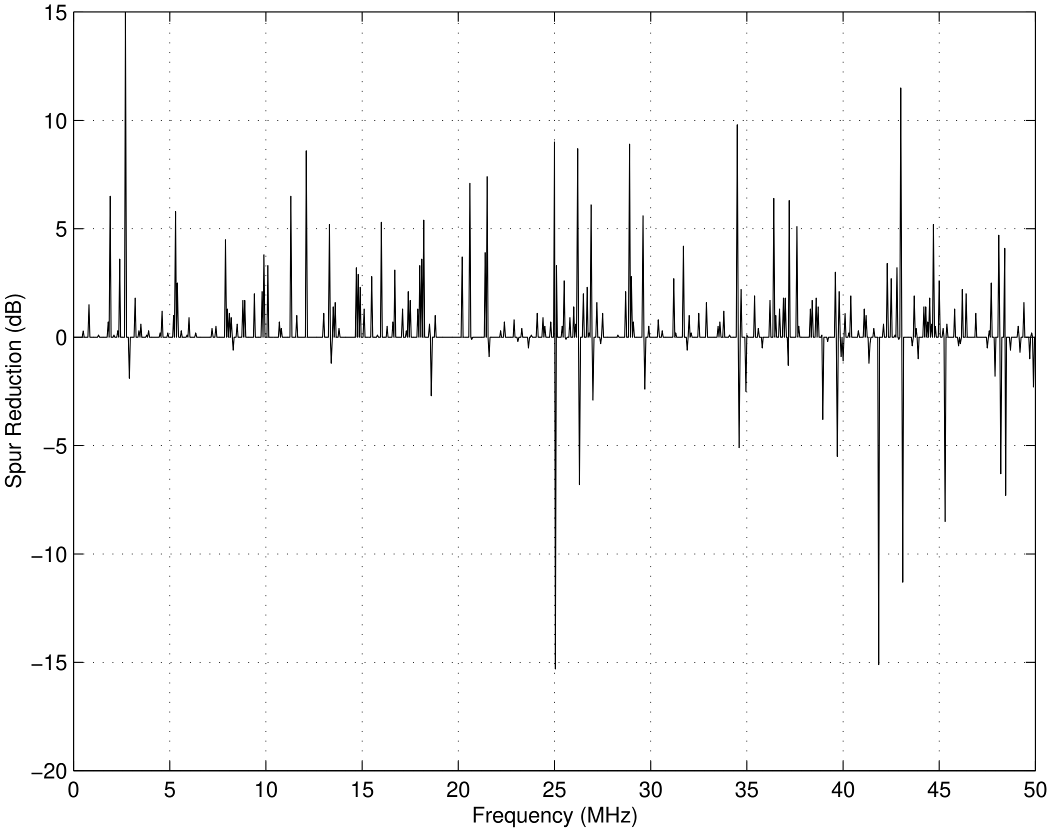

Figure 5.12: Spur reduction for $\alpha = 50$.

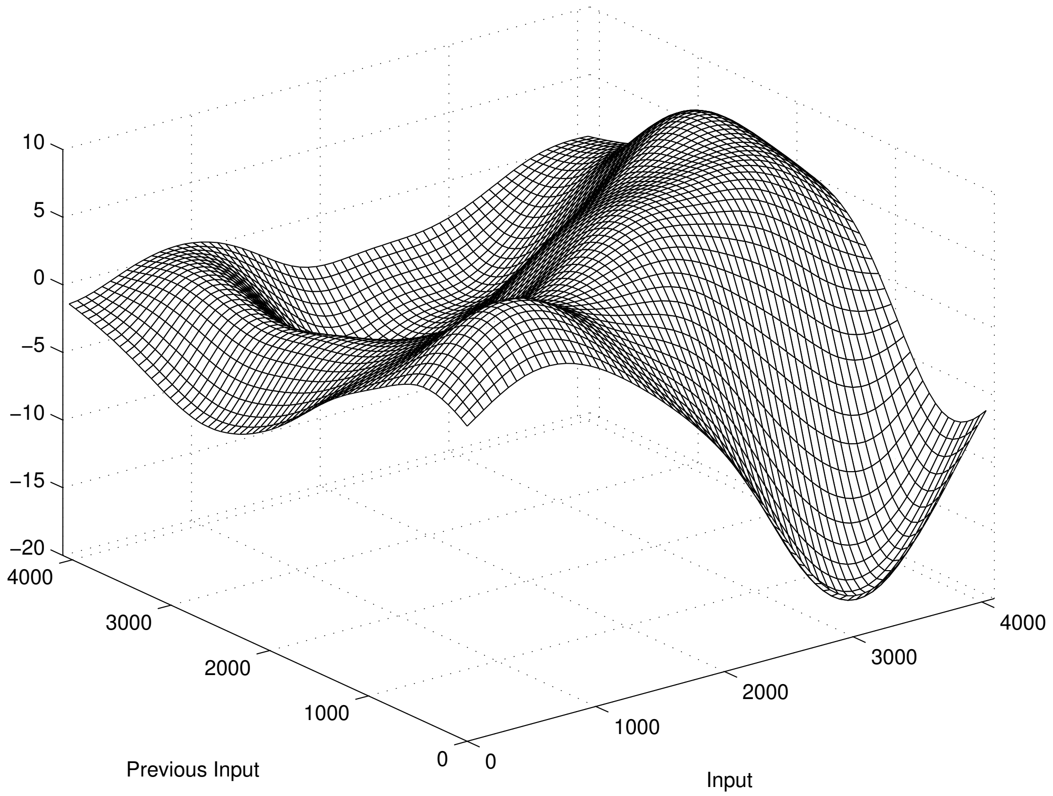

Figure 5.13: Error table for $\alpha = 50$.

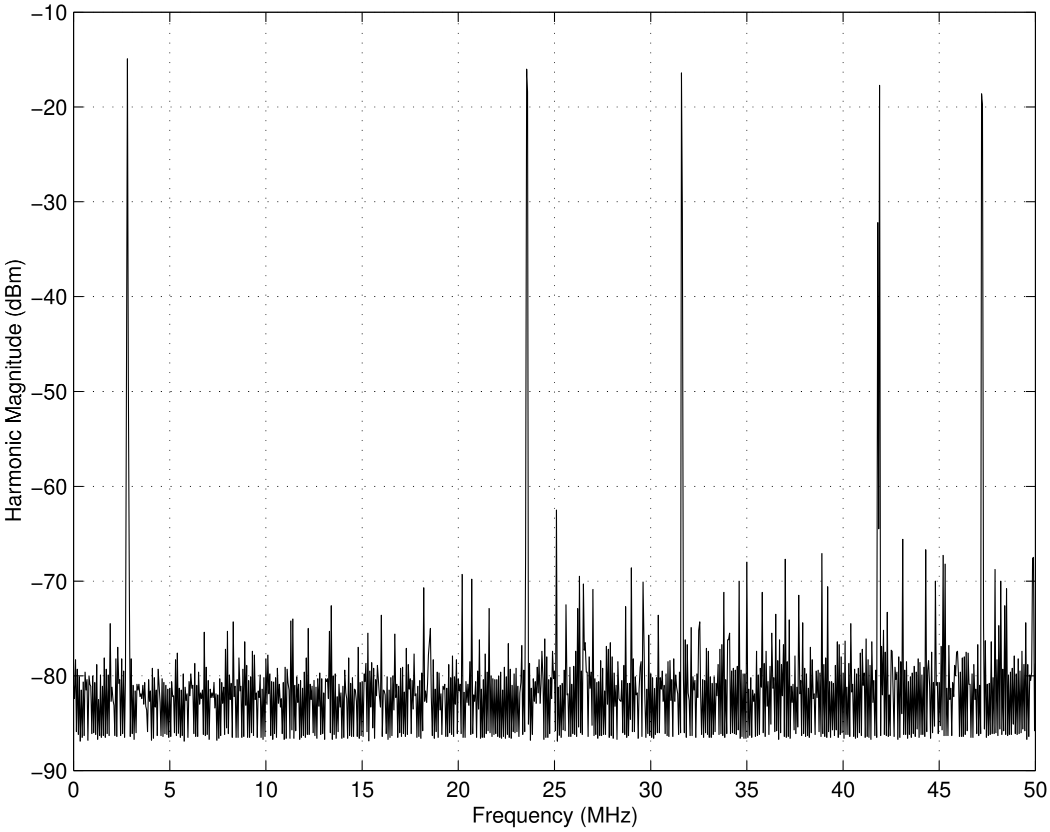

Figure 5.14: Compensated DAC output spectrum $\alpha = 400$.

The error table for α equal to four hundred is shown in Figure 5.15. Examination of this error table shows a shape similar to the $\alpha = 50$ error table, with smaller values. At this point the error table is becoming too limited and performance begins to suffer, as shown by more negative values in Figure 5.16. From these results it we conclude that a wide range of α values will produce similar results. α selection is best determined by comparative examination of the error table and output spectrum results for different α values.

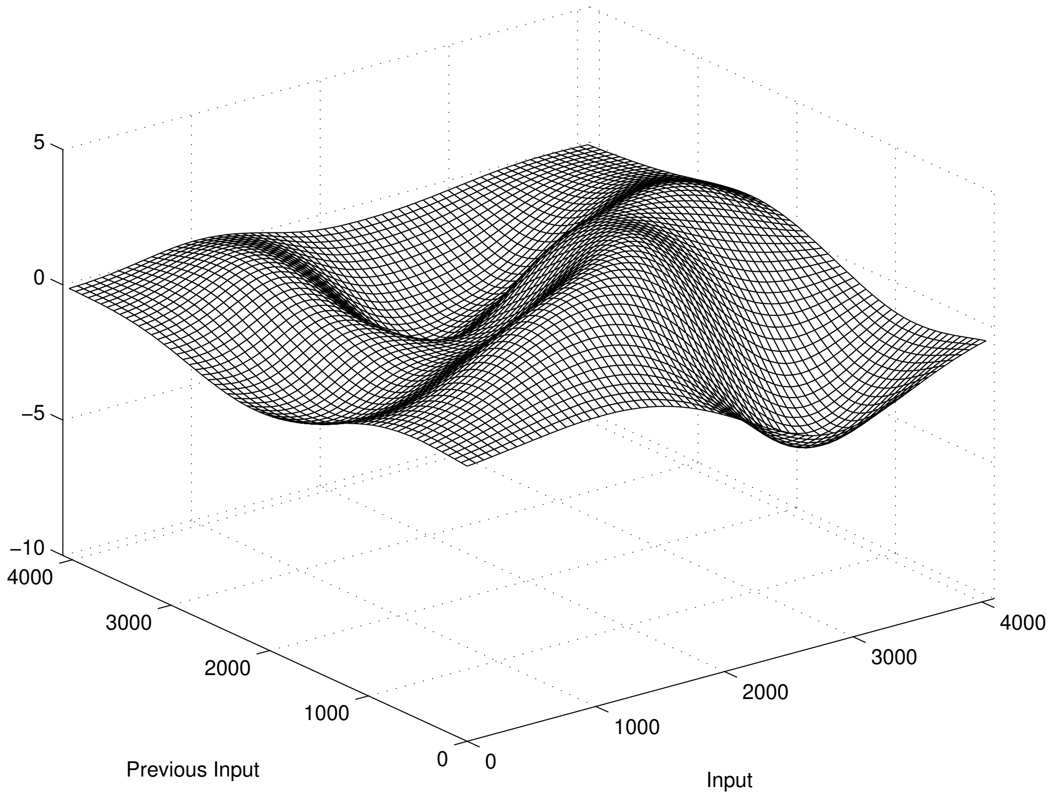

Figure 5.15: Error table for $\alpha = 400$.

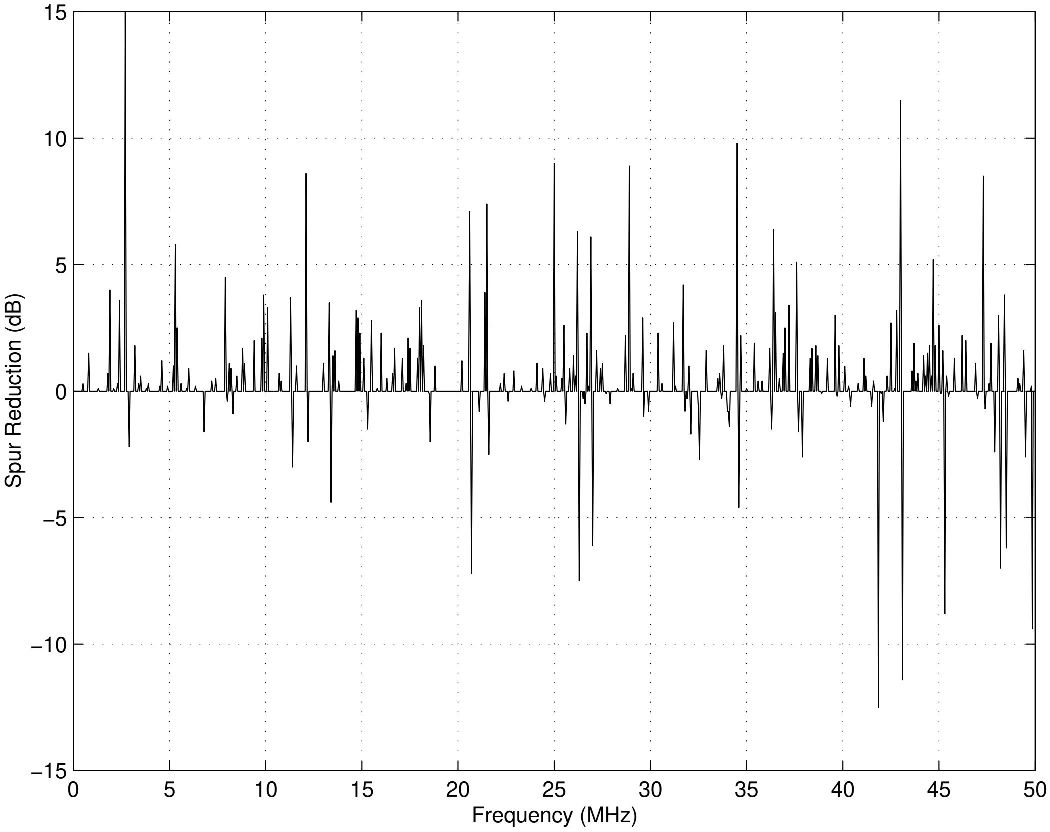

Figure 5.16: Spur reduction for $\alpha = 400$.

CHAPTER 6

Conclusions

This thesis has described techniques for calibrating and compensating DAC errors. A dynamic DAC error model based on the DAC current and previous inputs was developed (Chapter 2). This model sums an error signal with the input and applies the result to an ideal DAC input. The dynamic error model described is capable of modeling several different dynamic error mechanisms including slewing effects and bit-transition effects. A compensation method was developed using this DAC error model (Section 2.4).

A system for calibrating the DAC and determining the error has been investigated (Chapter 3). The error as a function of DAC input and previous input was estimated as the sum of a set of weighted basis functions of the DAC current and previous inputs. The basis function weights were determined by doing an LMS fit of estimated error to observed error in a limited bandwidth calibration spectrum.

This system was first evaluated using a simulated DAC with dynamic errors. It was determined that the calibration system as developed in Chapter 3 was incomplete and needed refinements described in Chapter 4. With these refinements it was shown that dynamic DAC errors may be compensated.

This system was then evaluated using a real DAC with dynamic errors. Using the calibration scheme developed in Chapter 4 and the compensation scheme described in Chapter 2, it was determined that errors that can be seen in the calibration spectrum will be compensated by this system. Errors that were not dependent on the DAC input and previous input (such as clock signal bleed-through and clock signal intermodulation) were not compensated by this system. Additionally, errors that were a function of current and previous inputs that had no energy in the calibration band were not compensated.

This thesis has presented a calibration and compensation scheme that is capable of compensating DAC errors that are functions of current and previous input. The measured results show that further work must be done in this area to produce a working dynamic calibration scheme for all error types. In particular, DAC error mechanisms have been identified which produce very little energy at low frequencies, and cannot be compensated by this lowpass calibration scheme. The next step in producing an effective dynamic calibration system is to provide the calibration process with a sampling of the higher frequency portion of the spectrum so that all error that is a function of current and previous input can be compensated. This may be accomplished by using a mixer to shift some high frequency error terms to baseband. This is an area of continuing investigation.

References

[1] Analog Devices. *Data Conversion Products Databook.*, 1989.

[2] Paul Hendriks. Specifying communication DACs. *IEEE Spectrum*, pages 58–69, July 1997.

[3] D.M. Hummels, I. Papantonopoulos, and F.H. Irons. Characterization of ADCs using a non-iterative procedure. In *Proceedings of IEEE International Symp. on Circuits and Systems*, London, May 1994.

[4] F. H. Irons and T.A. Rebold. Characterization of high frequency analog to digital converters for spectral applications – project report AST-2. Technical report, MIT Lincoln Laboratory, Lexington, Massachusetts, November 1986.

[5] S.P. Kennedy. Improved ADC phase plane error compensation. Master's thesis, University of Maine, Dept. of Electrical and Computer Eng., Orono Maine, May 1992.

[6] J. Larrabee. ADC compensation using a sinewave histogram method. Master's thesis, University of Maine, Dept. of Electrical and Computer Eng., Orono Maine, August 1997.

[7] Alan V. Oppenheim and Ronald W. Schafer. *Discrete-Time Signal Processing.* Prentice Hall, 1989.

[8] L.L. Scharf. *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis.* Addison-Wesley, Reading, Massachusetts, 1991.

Biography of the Author

Kirk Riley was born in Lewiston, Maine on November 10, 1973. He received his high school education from Jay High School in Jay, Maine in 1992.

He began study at the University of Maine in the fall of 1992 and received his Bachelor of Science degree in Electrical Engineering in August of 1996.

In the fall semester of 1996 he enrolled for graduate study in Electrical Engineering at the University of Maine and served as both teaching assistant and research assistant. His current research interests include data conversion and signal processing. He is a member of Eta Kappa Nu and his interests include fiction and music. He is a candidate for the Master of Science degree in Electrical Engineering from the University of Maine in August, 1998.