DESIGN CONSIDERATIONS FOR CHARACTERIZATION OF CAPACITORS

by

MICHAEL DELIBERO

Submitted in partial fulfillment of the requirements for
the degree of Master of Science

Thesis Advisor: Dr. Merat

Department of Electrical Engineering
& Computer Science

CASE WESTERN RESERVE UNIVERSITY

TBD

CASE WESTERN RESERVE UNIVERSITY

SCHOOL OF GRADUATE STUDIES

We hereby approve the master of science of

Michael DeLibero

candidate for the Master of Science degree*.

*We also certify that written approval has been obtained for any proprietary material contained herein.

Dedication

Dedication text

Table of Contents

Table of Contents i

List of Figures iv

List of Tables v

Preface vi

Acknowledgments vii

List of Abbreviations viii

Glossary ix

Abstract x

0.1 Previous Abstract x

0.2 Current Abstract x

1 Todo 1

1.1 MATLAB 1

1.2 Parameters 1

1.3 Regression 1

1.4 Schematic Explanation 1

1.5 Appendix 1

2 Background 2

3 History of Capacitors 3

4 Capacitor Parameters 6

4.1 Practical Capacitor Uses 7

4.1.1 Bypassing . . . 7
4.1.2 Analog Filtering . . . 8
4.1.3 DC Blocking . . . 8
4.1.4 Oscillators . . . 8
4.2 Capacitance . . . 9
4.3 Impedance . . . 10
4.4 Phase . . . 12
4.5 ESL . . . 12
4.6 ESR . . . 14
4.7 Resonance Frequency . . . 14
4.8 Dissipation Factor . . . 15
4.9 Quality Factor . . . 15
4.10 Leakage Resistance . . . 16
4.11 Dielectric Absorption . . . 16

5 Measurement Circuitry **17**

5.1 Power . . . 18
5.2 DC Bias . . . 18
5.3 Optocoupler . . . 19
5.4 Charging Circuitry . . . 20
5.5 Discharging Circuitry . . . 20
5.6 Current Measurement . . . 21
5.7 Filtering . . . 22
5.8 Magnitude . . . 23
5.9 Phase . . . 24
5.10 ADC . . . 24
5.11 Communications . . . 25

5.11.1 USB 25
5.11.2 RS-232 25

6 Regression Analysis and Modeling **25**
6.1 Regression Analysis 26
6.1.1 Basic LSE 26
6.1.2 Levy's Technique - Complex Curve Fitting 29
6.1.3 Weighted LSE 31
6.2 Modeling 32
6.2.1 6 Term Model 34

7 Conclusion **39**

8 Future Work **39**

A Schematic **40**

B Generating Modeling Images **52**
B.1 Example Data: Figure: 15 52
B.1.1 Capacitor Subcircuit Model 53
B.1.2 Plot ExCapData Script 54
B.2 Basic LSE Image: Figure: 16 55
B.3 Levy's Method: Figures: 17, 18, 19, & 20 56
B.4 Utility Functions 65

List of Figures

1 Power Supply Bypassing Circuit 7
2 Analog Filtering Circuit 8
3 DC Blocking Capacitor 8
4 Oscillator Circuit 8
5 Low Pass Filter – Varying C 10
6 Capacitor Magnitude Over Frequency 11
7 ESL Capacitor Model 12
8 Capacitor Impedance with ESL 13
9 ESR Capacitor Model 14
10 RLC Capacitor Model 15
11 Loss Tangent 16
12 Dielectric Absorption 17
13 Operating Area 21
14 Low-Pass Sallen-Key Filter [34] 22
15 GRM31MR71H105KA88 Capacitor Data 26
16 Basic LSE 28
17 Levy's Technique 31
18 LSE + Iteration – Magnitude and Phase Error 33
19 LSE + Iteration – Combined Error 33
20 LSE + Iteration 34
21 6 Term Model 34
22 6 Term Model: Bad Initilization 36
23 6 Term Model: Good Initilization 38
24 LTSpice Schematic for Capacitor Model 52

List of Tables

1 SRS-PS350 Analog Control Characteristics [39][38] 18

2 Current Measurement Ranges 21

3 Sallen-Key Filter Specifications 23

Preface

Preface text

Acknowledgments

This should be the acknowlegement

List of Abbreviation

List of Abbreviations text

Glossary

List of Abbreviations text

Abstract

0.1 Previous Abstract

This paper presents an analysis of capacitor performance at up to 600VDC bias. The analysis is accomplished through small-signal impedance testing, charge and discharge time constant measurements, and leakage current measurements. All tests are recorded by means of multiple ADCs and computer analysis.

0.2 Current Abstract

This paper presents an analysis of capacitor performance degradation at up to a 600VDC bias. It proposes a testing method through small-signal impedance testing, charge and discharge time constant measurements, and leakage current measurements. Circuit analysis and initial prototypes will be discussed.

1 Todo

1.1 MATLAB

- Combine iteration scripts into a single unit.

1.2 Parameters

- What model is best suited for titanium electrolytic capacitors?

1.3 Regression

- Include some discussion on J. Miller's modeling techniques for supercapacitors.
- Either add more models or don't say that you will in the intro
- Add nodes to 6-term model.

1.4 Schematic Explanation

- Create a flow chart for the schematic.
- Plot the safe operating area with Matlab.
- Talk about why this phase measurement technique was chosen over others and their comparable accuracy.

1.5 Appendix

- Add code for Levy's method.

2 Background

The following is a list of the questions that I will be answering in my background section.

1. Where are capacitors used with a high DC bias?
    (a) What characteristics are the most important there?
    (b) What are the main failure modes?
    (c) What are the current specifications of the parts in use now?
    (d) What research is being done to develop better capacitors for this use case?
    (e) How do they currently evaluate the capacitors?
    (f) How would they benefit from my research?
2. What is the state of the art in capacitance measurement?
    (a) Impedance analyzers
    (b) Capacitance bridges
    (c) Hi pot testers
    (d) What are the good and bad points of each of these technologies?
    (e) Why do they not solve the problem that I stated?
    (f) Why does this technology not currently exist?
3. What work has been done similar to this in the past?
4. What are the important characteristics of capacitors?
5. Is there any evidence that a capacitor's properties will change over DC bias?

3 History of Capacitors

This section will chronolog the history of capacitors. It will link various introductions in the technology to advances in industry, and it will map the driving forces behind capacitor development.

Capacitors have their origin in the invention of the Leyden jar by Peter van Musschenbroek of Leiden University in 1745 [16]. Before this point, scientists were able to generate static electricity through electrostatic machines, but had a very limited ability to store this electrical energy [13]. The most common design for the Leyden jar was to use a glass jar with metal foil lining the inside and out. The inner foil was typically charged via an electrostatic generator, while the outer foil was connected to ground. The charge would stay on the metal foil until a short or small resistance was connected between them. Charge could be stored this way, allowing scientists and showmen, to use greater amounts of current than they could generate at any one moment. Since the Leyden jar, many different types of capacitors have risen and fallen in prominence in the market. This section will cover the historical introduction of some of the major types.

In 1876 Fitzgerald introduced wax impregnated paper dielectric capacitors with foil electrodes [5, ch. 11][21]. They were typically used for power supply filtering in radios. By the early 1920s, they existed as tubes encapsulated in plain, Bakelite cardboard, with bitman sealing the ends and waxed paper as a dielectric [5, ch 3]. Paper capacitors were upgraded to impregnated paper capacitors, which used paper that was soaked in mineral oil. They were interleaved with metal foil and then rolled to make the capacitor [32, ch. 8.2.1.1]. During WW2, paper capacitors were upgraded with metal-cased tubes with a rubber end [32, ch. 8.1]. These metalized paper capacitors were constructed similarly with the improvement that one side of the paper dielectric was sprayed with metal [9].

Karol (Charles) Pollak discovered the principle of the electrolytic capacitor in 1886 while he was researching the anodizing of metals, and received a patent for the borax-solution aluminum electrolytic capacitor in 1897. In 1936 Cornell-Dubilier opened a factory to produce aluminum electrolytic capacitors. After the start of WW2, increased funding and effort was applied to the cause of electrolytic capacitors and techniques such as "etching and pre-anodizing" greatly increased their reliability [11][44].

The science of electrolytic capacitors was extending into what is now know as electrochemical capacitors in 1957 by GE. This technology, also known as electric double layer capacitors, was not commercialized until NEC licensed Standard Oil's patent in 1978 [26]. This "supercapacitor" formed the basis of today's EC capacitors. They are characterized by a large capacitance in the range of kilo farads, but with a working voltage of only 2.7V [26]. Some of today's typical applications for EC capacitors are in battery replacement, electric vehicle, and regenerative breaking.

M. Bauer of Germany invented the mica capacitor in 1874. The original mica capacitor was a "clamped" style capacitor, which was used through the 1920s [45] and then replaced by silver mica capacitors [21]. Mica's inherent inertness and reliability allowed for extreme reliability and efficiency in a packaged capacitor [40]. Mica capacitors were heavily used in the radio industry due to their superb stability at RF frequencies and their physical robustness [31]. Capacitors created with mica allowed a comparatively smaller product [5, f. 37-41] which had the ability to survive shock from weapons better than its glass counterpart. Consequently, mica capacitors began to be produced in large quantities during WWI.

In light of mica supply chain problems and the emergence of ceramic capacitors during WW2, mica capacitors fell from prominence to a niche market [2, Ch 3, Sec II].

The first glass ceramic dielectric capacitor was the Leyden jar. While this early capacitor was used mainly for scientific experiments, commercial glass capacitors

came later. Glass tubular capacitors, known as Moscicki tubes, appeared in 1904 and were used in Marconi's experiments in wireless transmission. They continued to be used in wireless communication until about WWI [5, p. 102].

Scientists in Germany created the first steatite ceramic capacitor in 1920 [2, Ch 3 Sec II][15]. Also known as talc, this ceramic capacitor variant was able to closely match the temperature coefficient of mica [10]. Rutile was later introduced into ceramic capacitor technology. It was able to produce a dielectric constant of 10 times that of steatite, but the two were typically blended to get a better temperature coefficient.

A ceramic composition with barium titanate (BaTiO3) was first discovered in 1941. Barium titanate was quickly found to be able to exhibit a dielectric constant over 1000; an order of magnitude greater the best at this time (rutile - $Ti0_2$). It was not until 1947, that barium titanate appeared in its first commercial devices, phonograph pickups [14][9][2, Ch 3 Sec III]. Barium titanate is still used today in certain types of multi-layer ceramic capacitors.

Thanks to the semiconductor industry, MLCCs grew to dominate the capacitor market and by 1979, AVX alone reached $95 million in MLCC sales [1]. MLCCs are divided up into three classes. Class 1 type MLCCs are known for their extremely good temperature characteristics. COG/NPO types can have 0-30ppm/^{o}C. These capacitors are typically made by combining TiO_2 with additives in order to adjust its temperature characteristics [28]. Additionally, class 1 ceramics have comparatively poor volumetric efficiency, and will tend to come in larger packages than class 2 ceramics of the same capacitance value. Class 2 types typically have worse temperature coefficients than type 1 types, but they have a much higher volumetric efficiency. They are constructed with a ferroelectric base material, typically Barium Titanate [28]. Class 2 ceramic capacitors are the most common type of ceramics used. Class 3 types have very high capacitance, but a working voltage of several volts [9][2, Ch 3 Sec VI][8]. They were originally developed as a potential replacement for liquid

electrolytics, but have fallen out of favor due to the advances in Class 2 ceramics to the point where the gap between these two in terms of maximum capacitance is no longer viable [43].

Bell labs invented the first solid tantalum capacitor in 1956. They created it in conjuncture with, and for, transistors [5, f. 56-64]. Tantalum capacitors typically have better characteristics than aluminum electrolytics, but have a lower maximum capacitance and working voltage [21]. Sprague patented the first commercially viable solid tantalum capacitor in 1960. It offered an increased capacitance per unit volume and greater reliability [33]. In the 1970s, Sprague released the first surface mount tantalum capacitor [37]. One of the historical problems with tantalum capacitors has been a limited and volatile supply of tantalum in the world market. As a result of a price spike around 1980, manufacturers created finer grain tantalum powders. This allowed a unit to be made with less overall tantalum, reducing price and package size [12, ch 3.1]. Over time, this volatility has decreased the popularity of tantalum capacitors and has led to a desire for a suitable replacement.

One of the possible alternatives to tantalum capacitors is a titanium based capacitor. Some of the instrumental work in titanium alloys was done in 1939 by Fast [7] and more recently in 1995 by Kobayashi [19][4]. Early work on titanium based capacitors resulted prohibitively high leakage currents [18], but research done by Welsch [42] with titanate (TiO_2) promises a capacitor with a much higher energy and power density when compared with tantalum [6].

4 Capacitor Parameters

This section will begin by describing some of the practical uses of capacitors and then transition into the various parameters that are used to describe capacitors. Section: 6 will show a method which allows these parameters to be extrapolated from empirical

data.

4.1 Practical Capacitor Uses

The most basic reason for wanting to use a capacitor is that it has the ability to store charge; it has the ability to store electrical energy. Capacitors have the ability to store and release electrical energy quickly, in order to be able to react to the needs of the circuit. This section will describe some of the most common uses for capacitors.

4.1.1 Bypassing

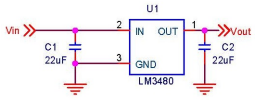


Figure 1: Power Supply Bypassing Circuit

One of the most common uses of capacitors is in power supply bypassing. Capacitors are nearly always attached from a power rail to ground on an supply or IC. They provide a resevior of charge that limits inductive voltage spikes, such as when a digital circuit switches, and limits voltage dips, caused by things such as a current surge when a processor boots. Both LDOs (Figure: 1) and switchers use bypass capacitors on their input and output voltage rails. Input capacitors are divided up into two main catagories, ripple reduction and bulk. Ripple reduction capacitors need to have a low ESR (Section: 4.6) and are meant to decrease the magnitude of any AC signals that ride ontop of the input DC voltage. Bulk capacitors are meant to deliver surge currents. Ouput capacitors have much the same purpose as the input capacitors. The main difference is in the case of switching power supplies. In that application, the output capacitor is a major component in the feedback loop. It contributes to both the transient and stability properties of the switcher.

4.1.2 Analog Filtering

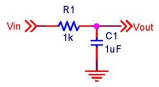


Figure 2: Analog Filtering Circuit

Another use for capacitors is in analog filtering. The low-pass filter in Figure: 2 attenuates frequencies above a cutoff point, set by the values of the resistor and capacitor. Low pass filters are needed in many applications, such as anti-aliasing, clock filtering, and integration.

4.1.3 DC Blocking

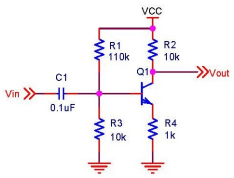


Figure 3: DC Blocking Capacitor

[29][ch 2.2.7 fig 2.35 pg 88]

Designers often take advantage of capacitors' characteristic of passing AC current while blocking DC current. As in Figure: 3, a capacitor can be used to block a DC offset before an amplifier.

4.1.4 Oscillators

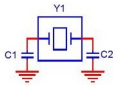


Figure 4: Oscillator Circuit

Stable capacitors of a very specific value are required to make a parallel resonant oscillator function properly (Fig: 4). These oscillators provide the clock base for most modern digital circuits using microcontrollers.

4.2 Capacitance

There is a distinct difference between a capacitor and capacitance. While a capacitor's dominant characteristic is capacitance, it cannot be modeled entirely as such in most practical applications. There are also various inductive and resistive components to a capacitor that are important in various circumstances.

$$C = \frac{Q}{V} \tag{1}$$

Capacitance is the ability to store electrical charge. Equation: (1) shows that capacitance is stored charge that is spread throughout a volume. A device that can store a lot of charge in a small space has a large capacitance. The basic equation for a commercial capacitor is seen in Equation: (2).

$$C = \frac{\epsilon_0 A}{d} \tag{2}$$

When using a capacitor in a single-pole low-pass filter, the cutoff frequency can be determined by Equation: (3). The circuit designer will choose a value for C and R in order to meet the cutoff frequency restraint.

$$f = \frac{1}{2\pi RC} \tag{3}$$

Varying the capacitance used in the filter will move the cutoff frequency and consequently get a different response in the filter. The effect of this can be seen in Figure: 5.

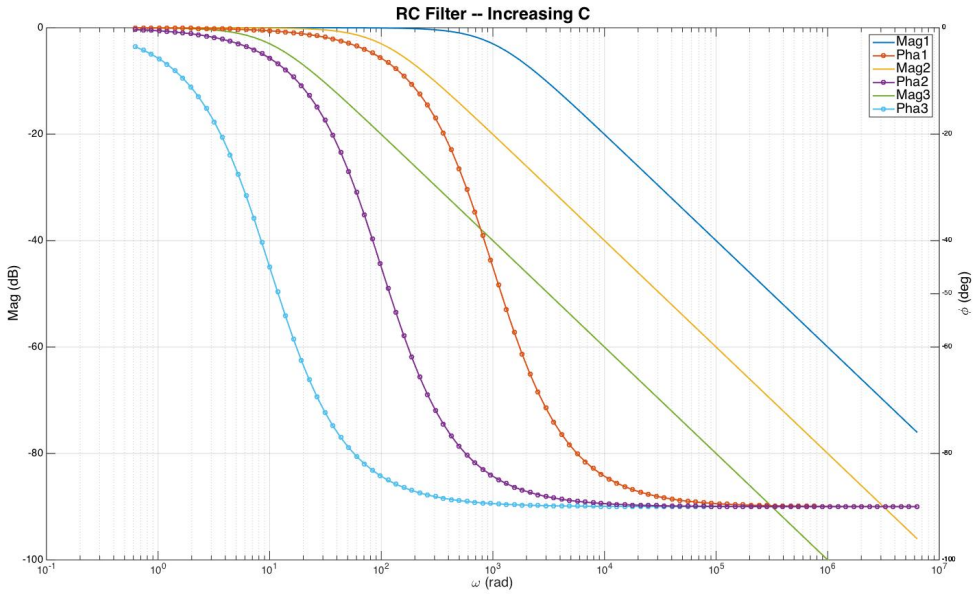


Figure 5: Low Pass Filter – Varying C

4.3 Impedance

The impedance of a capacitor is the "AC resistance" of the device. It determines the AC current that will flow when an ac voltage is applied to the capacitor via Ohm's law (Equation: (4)). An ideal capacitor has only a single capacitive element and its impedance can be described via Equation: (5). The two main things to notice are that the impedance is frequency dependent and it is purely imaginary (reactive).

$$\vec{V} = \vec{I}\vec{Z} \tag{4}$$

$$\vec{Z} = \frac{1}{j\omega C} \tag{5}$$

$$Z = |\vec{Z}| = \frac{1}{\omega C} \tag{6}$$

In most AC applications we look at the magnitude of the impedance. Real capacitors

have a more complicated impedance, but with an ideal capacitor we can simplify the magnitude equation down to Equation (6)

When capacitors are used in bypassing power supplies, the idea is to have a low impedance for common or expected noise frequencies. One may be tempted to choose a large valued capacitor to use for bypassing a wide range of frequencies. This turns out to backfire in practical situations, due to other parasitics in a real capacitor. For any capacitor, the impedance equation is more complicated, and the impedance value will begin to increase with frequency after some point. This will cause the designer to choose several different valued capacitors in parallel when bypassing a power supply or sensitive component. We will see later that the frequency plot of a capacitor will end up being more complicated than the simplified version seen in Figure: 6.

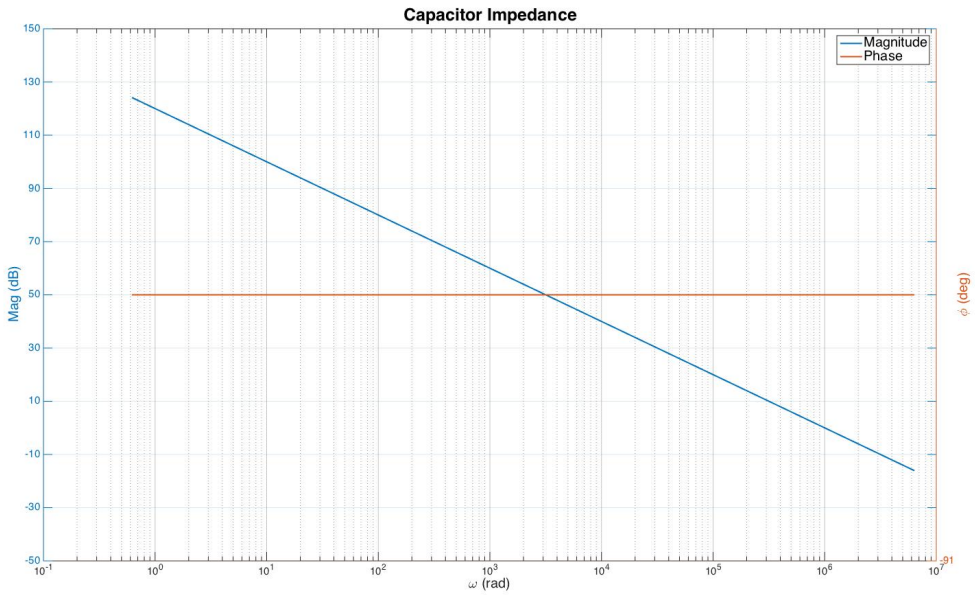


Figure 6: Capacitor Magnitude Over Frequency

4.4 Phase

The phase of a combination of resistive and reactive components can be written as in Equation: (7).

$$\phi = tan^{-1}[\frac{X_c}{R_c}] \tag{7}$$

For an ideal capacitor, having no resistance and only capacitance, the phase angle can be simplified to:

$$\phi = -i = -90^0 \tag{8}$$

The practical implication of this can be seen in the phase response of a low pass filter (Figure: 5). The capacitor introduces a phase lag relative to the input signal's frequency. If you would compare the input and output signals in time, the output's peak would lag behind the input's by the phase amount predicted in the phase response.

4.5 ESL

The Equivalent Series Inductance (ESL) of a capacitor is a lumped estimate of all of the inductive components of a capacitor. It is typically modeled as an inductor in series with the bulk capacitance (See Figure 7).

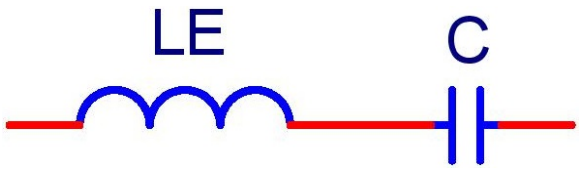


Figure 7: ESL Capacitor Model

Adding ESL to the capacitive model creates a new impedance equation (Equation: (9)). Note that for L<<C, this equation simplifies to Equation: (5) for low

frequencies. In other words, the ideal impedance equation can be reasonably used for low frequencies.

$$\vec{Z}_c = j\frac{\omega^2 LC - 1}{\omega C} \tag{9}$$

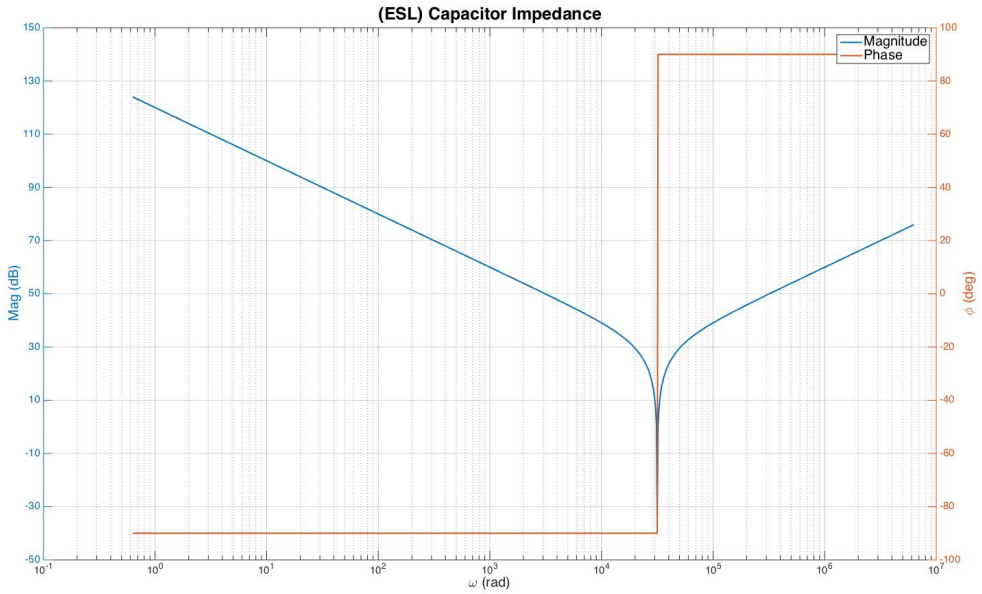


Figure 8: Capacitor Impedance with ESL

Figure: 8 shows a graphical representation of a capacitor's magnitude and phase once ESL is considered. This plot shows that after a resonance point, the impedance of the inductor (which increases with frequency) will begin to dominate. This makes the capacitor ineffective as a bypass element at frequencies higher than its resonance point. Typically, this frequency point and the capacitor's value have an inverse relationship. This is why you will see power supplies and other chips being bypassed by a range of different valued capacitors.

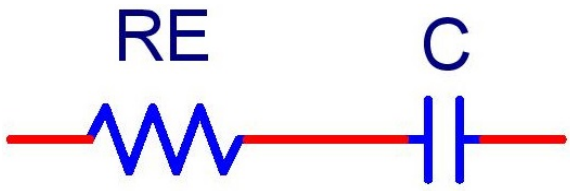


Figure 9: ESR Capacitor Model

4.6 ESR

The Equivalent Series Resistance (ESR) is the practical result of the fact that the materials used to create a capacitor have resistance. In simple cases, this can be approximated by a resistance in series with the main capacitor (See Figure: 9).
ESR becomes is important when thinking about DCDC switch mode power supplies. The output ripple voltage of the converter will cause a rripple current to pass through the ESR and dissipate heat as per Equation: (10). It is important to choose a low ESR capacitor in order to reduce failures.

$$P_E = I_{C,RMS}^2 * R_E[24] \tag{10}$$

Another important thing to note about ESR is that even though it is shown as a resistance in simple models, it is not constant across all frequencies. It is a simplification of the resistive and capacitive elements in a capacitor that are dominated by resistance. That said, it is sufficient for a basic understanding of a capacitor's impedance (Equation (11).

$$\vec{Z_c} = \frac{1 + j\omega R_E C + (j\omega)^2 L_E C}{j\omega C} \tag{11}$$

4.7 Resonance Frequency

Once C, ESL, and ESR are included into the capacitor model (Figure: 10, a parameter known as the self-resonant frequency becomes evident. Equation: (11) shows that

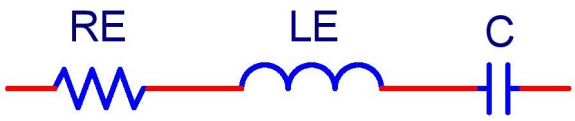


Figure 10: RLC Capacitor Model

when $Z_{ESL} == Z_C$, the capacitor is at its resonance point. At this frequency, the capacitor's impedance is determined solely by the ESR at that frequency. This frequency can be calculated by Equation: (12).

$$f_r = \frac{1}{2\pi\sqrt{LC}} \tag{12}$$

4.8 Dissipation Factor

The dissipation factor, otherwise known as the loss-tangent, is a measure of the energy stored to the energy dissipated per cycle. It is a measurement of the efficiency of the capacitor. The DF can be quantified through Equation: (13).

$$D = \frac{R_E}{X_C} \tag{13}$$

The loss tangent can be seen in Figure: 11. The greater the angle, the more efficient the capacitor will be.

4.9 Quality Factor

$$Q = \frac{1}{D} \tag{14}$$

The Quality Factor, Q, of a capacitor is found by taking the reciprocal of the dissipation factor, Equation: (14). It is defined as the ratio of the energy stored to the energy dissipated per cycle.

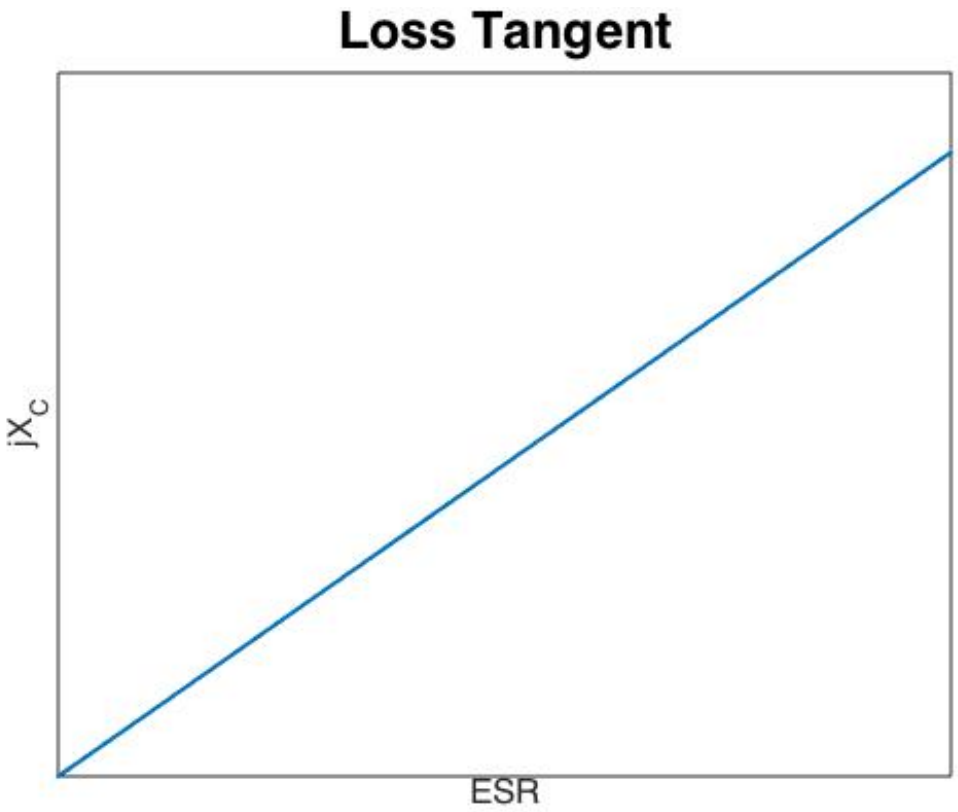


Figure 11: Loss Tangent

4.10 Leakage Resistance

Every capacitor will have some DC leakage resistance associated with it. This resistance affects the capacitor's ability to store charge. A high leakage resistance results in a capacitor with low self-discharge. This characteristic is especially important in sample and hold circuits.

4.11 Dielectric Absorption

Dielectric Absorption, DA, in a capacitor is a characteristic which describes the unit's ability to "regenerate" a voltage after being shorted to ground for a brief time.

As seen in Figure: 12, a capacitor can be modeled with multiple RC elements in

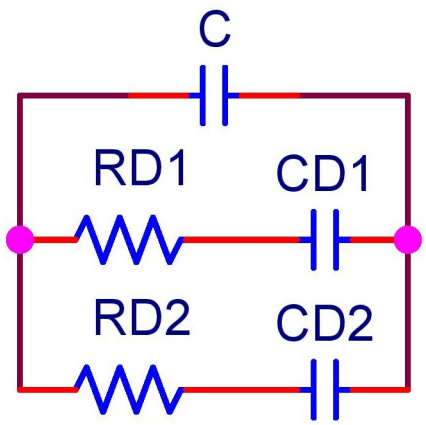


Figure 12: Dielectric Absorption

parallel with the bulk capacitance. When the main capacitor is shorted to ground for a short time, and then released, the other capacitors are not gauranteed to have been fullly discharged. After several minutes, they can recharge the main capacitance to a significant portion of its original charge. This is why large valued electrolytic capacitors get shipped with a resistor across their terminals.

5 Measurement Circuitry

This section describes the schematic design used to meet the goals set out in the Abstract (Section: 0.2). The schematic can be found in the Appendix: A. At the time of this writing, this circuit has not been constructed, so what follows will be a theoretical explaination of the circuit's operation.

The purpose of this circuit is to measure the response of a capacitor to various test inputs in order to generate a model (Section: 6). This method will allow an analysis of the effect of the DC bias on the output model.

5.1 Power

The power supply section, shown on Schematic Page: 2, shows the complete power generation scheme for the circuit board. 24VDC is fused and then fed through an EMC compliance filter [20]. The initial fuse value is set to roughly twice the current rating of the switching power supplies. Once a practical current draw value is obtained, this value can be decreased. The resistors on the output of the 2 switching regulators are there to ensure that the minimum load (10%) for the power supplies is met. The supplies require this in order to meet their regulation specifications. All of the power supplies where chosen such that their power atings were at least twice the maximum, expected load. The power supply generating $\pm 15V_{ISO}$ is slightly different than the rest, as its output ground is attached to the DC bias voltage instead of signal ground. This purpose of this seen in the optocoupler circuitry described in Section: 5.3.

5.2 DC Bias

The DC bias signal is externally generated by a Stanford Research Supply SRS-PS350 power supply. Some of the models of this supply provide a serial input which can be used to set their output voltage. This scenario is preferred and the communications for it are handeled through the RS232 transceiver found on Schematic Page: 3.

Input Scale	0 to +10V for 0 to 5kV
Input Impedance	1 MΩ
Accuracy	±0.2% of full scale
Update Rate	15 Hz
Output Slew Rate	<0.3s for 0 to full scale (full load)

Table 1: SRS-PS350 Analog Control Characteristics [39][38]

Other models of this supply only provide an analog input option to control their output voltage. The specifications for this method can be seen in Table: 1. The control signal is generated by using the microcontroller to set the DAC output over a SPI bus. This signal is then buffered and clamped to under 1.5V. Since the SRS-PS350 has a gain of 500, this clamps its output to 750VDC. The circuit is only designed to operate at up to 500VDC, but this value is still within the specified ratings of the components in the signal path.

The DAC has a 2.5V internal reference with a 14 bit resolution, giving the control signal an ideal resolution of $150\mu V$ and the SRS-PS350 an ideal output resolution of $76mV$. The update rate of this device is not important to this application because the DAC will be set infrequently and only when the system is not actively collecting data.

If different supply characteristics are needed, the SRS-PS340 can be swapped out with a different supply, but the control logic may not be supported.

5.3 Optocoupler

This section describes the circuitry shown on Schematic Page: 4. It is based upon IXYS's application note AN-107 [23], and its purpose is to inject an AC signal ontop of a high DC bias. The optocoupler is configured in what is known as photovoltaic mode [23][Sec: 3.3], with both photoreceivers acting as coupled current sources.

In this mode, the front end op-amp's feedback path causes the first photoreceiver to draw current through the first resistor such that both of its terminals stay and ground potential. The phototransmitter's resistor sets a full scale current of 20mA, which is well within the absolute maximum ratings of the device. The second phototransmitter pulls current through the the backend op-amp's feedback resistor with a voltage gain of 2.5.

The output of the optocoupler is referenced to the DC bias voltage from the SRS-

PS350 istead of to circuit ground. The limiting isolation specification comes from the $\pm 15V$ supply at 1.5kV, far below the 500V limit imposed by the system. The input sine wave is fed into the magnitude and phase measurement circuitry seen on Schematic Page: 7. The optically coupled sine wave is fed into the DUT as seen on Schematic Page 5.

5.4 Charging Circuitry

This section describes the DUT charging circuitry seen in Schematic Page: 5. This part of the circuitry is meant to only be used to prepare the DUT for a test. The current is purposely limited in order to minimize the load on the SRS-PS350. The charging operation functions by enabling the high-current measurement and charging relays, and then slowly ramping the DC bias voltage until the DUT is fully charged.

5.5 Discharging Circuitry

This section describes the operation of the discharging circuitry seen on Schematic Page: 5. It provides a means to determine the bulk capacitance value of the DUT from the RC time constant. Once the DUT is charged via the operation described in Section: 5.4, the charging relay will open and the high-current discharge relay will close. The transimpedance amplifier described in Section: 5.6 will measure the current through the DUT as it decays.

There are 3 switched discharge stages that allow for the decay rate to be regulated. Each stage allows for a range of voltages and capacitances needed to stay within safety and operational constraints. As seen in Figure: 13, there are two constraints on both the voltage and capacitance. The lowest valued resistor constrains the maximum voltage when using that stage due to its power rating. The other two stages are constrained by the overall safety limit of 500VDC. The minimum capacitance for

each stage is determined by setting a requirement of at least 100 ADC samples per time constant with a sampling rate of $250KSPS$. The maximum capacitance for each stage is determined by setting a maximum time constant.

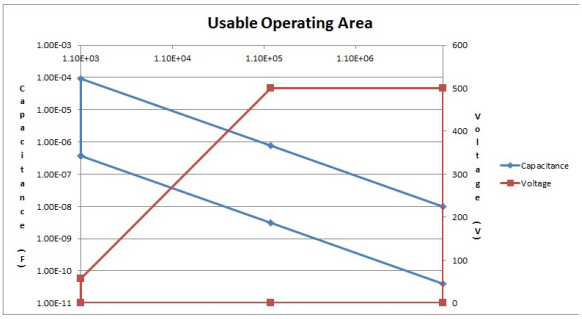


Figure 13: Operating Area

5.6 Current Measurement

This section describes the current measurement circuitry seen on Schematic Page: 5). In light of the high DC bias voltage present in the system (Section: 5.2) this circuitry utilizes a low-side current measurement technique. This allows the measurement electronics to operate at relative low (5VDC) voltages while measuring current from the DUT with a DC voltage in the 100s of volts.

Value	Range	Current
5	Hi	1A - 1mA
5k	Med	1mA - 1uA
5M	Low	1uA - 1nA

Table 2: Current Measurement Ranges

The current measurement circuit utilizes a transimpedance amplifier designed from the reference circuit in [6]. It utilizes 3 switched, feedback stages to measure 9 decades of current (See Table: 2). The circuit operates by creating a virtual ground at the

negative terminal of the DUT, and then directing the current through one of its feedback paths. The resultant voltage (Equation: (15)) is then filtered and sent to the next stage for further signal conditioning.

$$Vo = I * R_f \tag{15}$$

The AD817 op-amp has the precision and bandwidth needed to operate in the transimpedance amplifier, but not the drive strength for th high current range. This limitation is overcome by inserting a BJT based current booster in the feedback path. This preserves the benefits of the op-amp, while providing the ability to drive a high-current signal.

5.7 Filtering

This section describes the current filtering circuitry shown in Schematic Page: 6. Both filtering paths implement a 6-pole, low-pass, Sallen-Key filter to condition the signal.

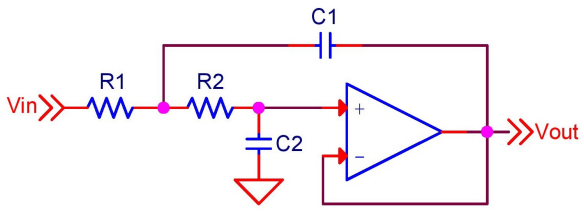


Figure 14: Low-Pass Sallen-Key Filter [34]

$$f_c = \frac{1}{2\pi R_2 C_2 \sqrt{\frac{R_1 C_1}{R_2 C_2}}} \; [29][ch\ 6.3.2.D\ equ\ 6.5\ pg\ 410] \tag{16}$$

$$Q = \frac{\sqrt{\frac{R_1 C_1}{R_2 C_2}}}{1 + \frac{R_1}{C_1}} \; [29][ch\ 6.3.2.D\ equ\ 6.6\ pg\ 410] \tag{17}$$

A genaric, single-stage, low-pass, Sallen-Key filter is shown in Figure: 14. The filter is basically a 2-pole, passive, RC filter with the ground of C_1 tied to V_{out} through the op-amp's active feedback path. This implementation is a simplified version of the Sallen-Key filter due to the gain (listed as K in most explanations) is equal to 1. The transition frequency and quality factor can be set according to Equations: (16) and (17) respectively by choosing values for the passives.

The filters for this circuit were designed with TI's WEBENCH designer [41]. Table: 3 shows the specifications for each filter.

The AC current measurement branch has a cutoff frequency of 100kHz, which gives

Filter	**Filter Order**	**Gain**	**f_c**	**Stopband Attenuation**
Discharging	6	1	25kHz	-65dB
AC Measurement	6	1	100kHz	-100dB

Table 3: Sallen-Key Filter Specifications

5.8 Magnitude

This setion describes the magnitude circutry shown on Schematic Page: 7. The circuit measures both the magnitude of the input sine wave and the magnitude of the resultant current waveform through the DUT. The design for this circuit is based off an Analog Device's app not for the AD8277 [27].

The function of this circuit is perform a full-wave rectification operation on the input signal and then pass the output to be filtered and then sampled. The first difference amplifier is configured as a voltage follower. Since it is only powered from a single rail, it passes the positive half of the waveform and shunts the negative half to ground. The second difference amplifier operates in different modes, based upon the output of the first differential amplifier. During the negative going portions of the waveform, its positive terminal is held at ground, and it operates as a unity gain, inverting

amplifier. This rectifies the input signal to the output. During the positive going portions of the waveform, positive input terminal is held at the value of the input terminal. This forces the second difference amplifier to act as a voltage follower. The resultant waveform is a full-wave recified version of the input signal.

5.9 Phase

This setion describes the phase circutry shown on Schematic Page: 7. The circuit measures both the phase of the input sine wave and the phase of the resultant current waveform through the DUT. It uses two, redundant phase measurement techniques. The first technique utilizes an RF phase detector (AD8302) in a low frequency configuration as per AN-691 [30]. The capacitors on the inputs tune the front end high pass filter of the IC to roughly 20Hz. The phase measurement is accomplished internally by a magnitude independent phase multiplier configuration. The output conisists of a 10mV per degree signal that is lowpassed at 2Hz to prevent aliasing. The filtered signal is then fed to the ADC for digitization.

The second technique is added for a low frequency comparison against the RF phase detector. The comparator acts as a sine to square wave converter. The default configuration does not include hysteresis, but provisions for it are available if needed. The outputs of the two converters are fed into timer inputs on the microcontroller. The phase is then calculated by measuring the period of the signals and the time between their rising edges.

5.10 ADC

This section describes the ADC circuitry found on Schematic Page: 8. It makes use of the AD7656, a 6 channel, 250ksps, 16bit, SAR ADC. It sends the converted values to the microcontroller for further processing and analysis via 3 parallel SPI outputs.

5.11 Communications

This section will describe the circuitry used to communicate to off-board processors.

5.11.1 USB

The USB section is used to communicate with a PC for data logging and post processing. This circuitry centers around a FTDI FT232H serial to USB chip. It allows seamless USB communication to a PC's com port with the MCU only needing to use a UART port. This significantly lowers the complexity of the communication bus, as the MCU is not responsible for running the USB stack.

5.11.2 RS-232

The board also has two, bidirectional RS-232 ports. They are able to be used for general purposes, but will most often be used for communicating with a sine-wave generator and the DC Bias supply.

6 Regression Analysis and Modeling

Many designers have a need to simulate the response of the individual components in their systems to impulse and steady state inputs. When dealing with passive electronic components, the device is typically modeled as a combination of various resistors, capacitors, and inductors. The model chosen is often due to either the required accuracy or a specific characteristic of the component that needs to be modeled. When characterizing a new component, the complex frequency response is recorded and then fit to a model. In this section, a progression of regression techniques will be evaluated for their ability to fit the measured frequency response of a capacitor to a polynomial equation. Then the accuracy of various capacitor models will be explored in regards to the fit. All models will attempt to fit the

capacitor impedance data shown Figure: 15. See Appendix: B for the code used to generate the images shown in this section.

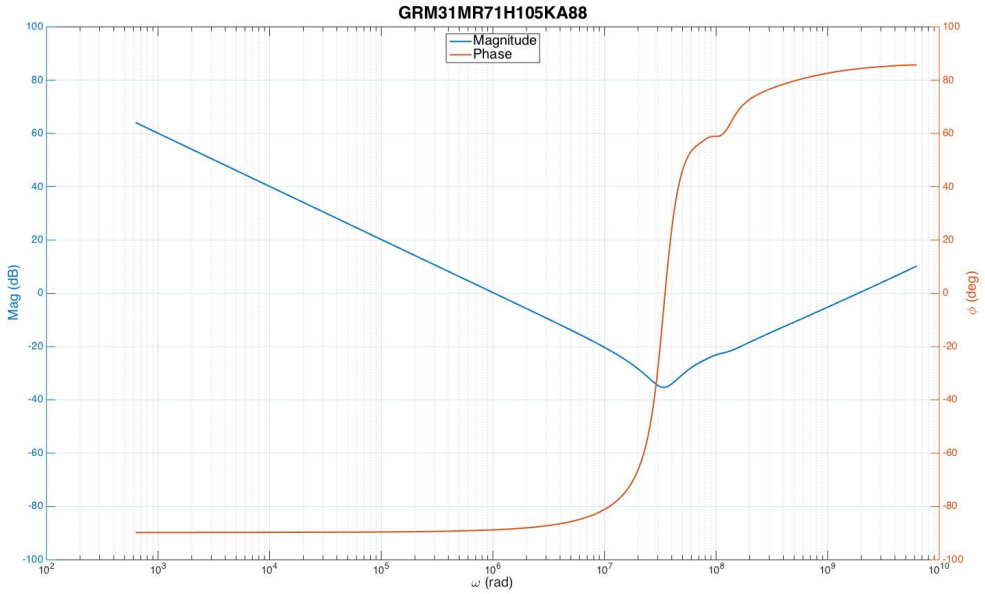


Figure 15: GRM31MR71H105KA88 Capacitor Data

6.1 Regression Analysis

6.1.1 Basic LSE

At its core, regression analysis is an optimization problem whose purpose is to fit the equation of a line to a data set. A commonly use regression analysis technique is called the Least Squares Estimate (LSE). It attempts to find a model which minimizes the squared error between an empirical set of data and itself.

The first step in applying an LSE is to choose the form of the equation that best represents the data. The equation of a line, Equation: (18), is chosen when only a simple linear fit is needed. Then the squared error equation is generated, as in

Equations: (19) & (20).

$$y = a_0 + a_1 x \tag{18}$$

$$E^2 = \sum_{i=1}^{n} (y_i - y)^2 \tag{19}$$

$$E^2 = \sum_{i=1}^{n} (y_i - (a_0 + a_1 x_i))^2 \tag{20}$$

In order to minimize the squared error over the data set, one needs to take the partial derivate of Equation: (20) with respect to each of the unknown parameters, the coefficients, separately. While a_0 and a_1 will be constants in the final equation, they are treated as variables here until they are known. Conversely, all x_i values are treated as constants. This results in Equations: (21) & (22).

$$\frac{\partial E^2}{\partial a_0} = 0 = \sum_{i=1}^{n} (-2y_i + 2a_0 + 2a_1 x_i) \tag{21}$$

$$\frac{\partial E^2}{\partial a_1} = 0 = \sum_{i=1}^{n} (-2y_i x_i + 2a_0 x_i + 2a_1 x_i^2) \tag{22}$$

Up to this point most LSE analyzes follow the same basic path. But the rest of the steps depend upon the complexity of the model and solution techniques. The following steps in the basic LSE use transformations and substitutions to solve for the unknown variables. In this case case, Equation: (23) is used to remove the summation terms from the equation.

$$\sum_{i=1}^{n} y_i = \bar{y} n \tag{23}$$

This results in Equations: (24) & (25) with solutions shown in Equations: (26) & (27). The empirical data is then used to find the values of a_0 and a_1. At this point,

the line can be used to estimate new points on the plot or to compare against other data sets.

$$0 = \bar{y} - (a_0 + 2a_1\bar{x}) \tag{24}$$

$$0 = \bar{xy} - (a_0\bar{x} + 2a_1\bar{x^2}) \tag{25}$$

$$a_0 = \bar{y} - a_1\bar{x} \tag{26}$$

$$a_1 = \frac{\bar{xy} - \bar{x}\bar{y}}{\bar{x^2} - \bar{x}^2} \tag{27}$$

An example of this type of fit can be seen in Figure: 16.

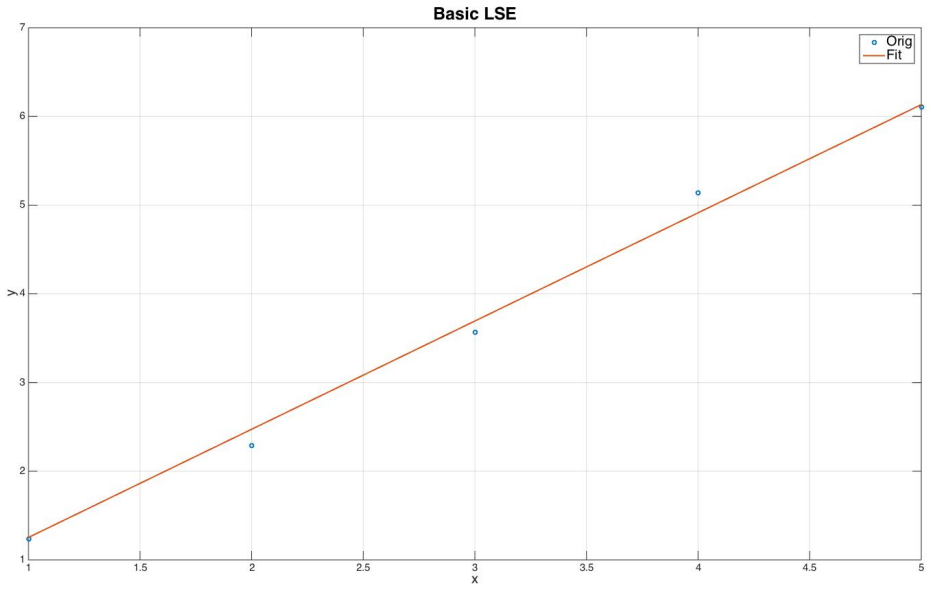


Figure 16: Basic LSE

6.1.2 Levy's Technique - Complex Curve Fitting

While the basic LSE technique is sufficient for many circumstances, it is not directly applicable in situations where one needs to fit a model to a complex line, such as a transfer function. Levy [22] shows an extension of the simple LSE example that is valid for a generic polynomial transfer function. This method is important because it not only allows for a complex-valued transfer function, but it also prevents the necessity of needing to rederive the system of equations for each new model.

$$G(s) = \frac{A_0 + A_1 s + A_2 s^2 + ... + A_n s^n}{B_0 + B_1 s + B_2 s^2 + ... + B_m s^m} \text{ [22][Eq. 3]} \tag{28}$$

Using Equation: (28) as the generic model, Levy shows that you can use Equations: (29), (30), (31), & (32) to simplify the series of partial derivatives into a single matrix multiplication equation shown in (33), (34), (35), & (36).

$$\lambda_h = \sum_{k=0}^{m} \omega_k^h \text{ [22][Eq. 15]} \tag{29}$$

$$S_h = \sum_{k=0}^{m} \omega_k^h R_k \text{ [22][Eq. 16]} \tag{30}$$

$$T_h = \sum_{k=0}^{m} \omega_k^h I_k \text{ [22][Eq. 17]} \tag{31}$$

$$U_h = \sum_{k=0}^{m} \omega_k^h (R_k^2 + I_k^2) \text{ [22][Eq. 18]} \tag{32}$$

$$MN = C \text{ [22][Eq. 20]} \tag{33}$$

$$M = \begin{bmatrix} \lambda_0 & 0 & -\lambda_2 & 0 & \lambda_4 & \cdots & T_1 & S_2 & -T_3 & -S_4 & T_5 & \cdots \\ 0 & \lambda_2 & 0 & -\lambda_4 & 0 & \cdots & -S_2 & T_3 & S_4 & -T_5 & -S_6 & \cdots \\ \lambda_2 & 0 & -\lambda_4 & 0 & \lambda_6 & \cdots & T_3 & S_4 & -T_5 & -S_6 & T_7 & \cdots \\ 0 & \lambda_4 & 0 & -\lambda_6 & 0 & \cdots & -S_4 & T_5 & S_6 & -T_7 & -S_8 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots & \vdots & \\ T_1 & -S_2 & -T_3 & S_4 & T_5 & \cdots & U_2 & 0 & -U_4 & 0 & U_6 & \cdots \\ S_2 & T_3 & -S_4 & -T_5 & S_6 & \cdots & 0 & U_4 & 0 & -U_6 & 0 & \cdots \\ T_3 & -S_4 & -T_5 & S_6 & T_7 & \cdots & U_4 & 0 & -U_6 & 0 & U_8 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \end{bmatrix} \quad [22][Eq.\ 21a] \tag{34}$$

$$N = \begin{bmatrix} A_0 \\ A_1 \\ A_2 \\ A_3 \\ \vdots \\ B_1 \\ B_2 \\ B_3 \\ \vdots \end{bmatrix} \quad [22][Eq.\ 21b] \tag{35}$$

$$C = \begin{bmatrix} S_0 \\ T_1 \\ S_2 \\ T_3 \\ \vdots \\ 0 \\ U_2 \\ 0 \\ \vdots \end{bmatrix} \quad [22][Eq.\ 21c] \tag{36}$$

Levy's technique works well for applications where there is a small dynamic frequency range and a small number of coefficients, but there a several problems with it. The first problem is that for models with a wide bandwidth, the solution to Equation: (33), involves an ill-conditioned matrix. This means that the solution to the system of equations will experience precision errors. The second problem is that this technique favors the magnitude plot at the high frequency range. As can be seen in Figure: 17,

a applying Levy's technique with 3^{rd} order model provides an unuseable fit to the emperical data.

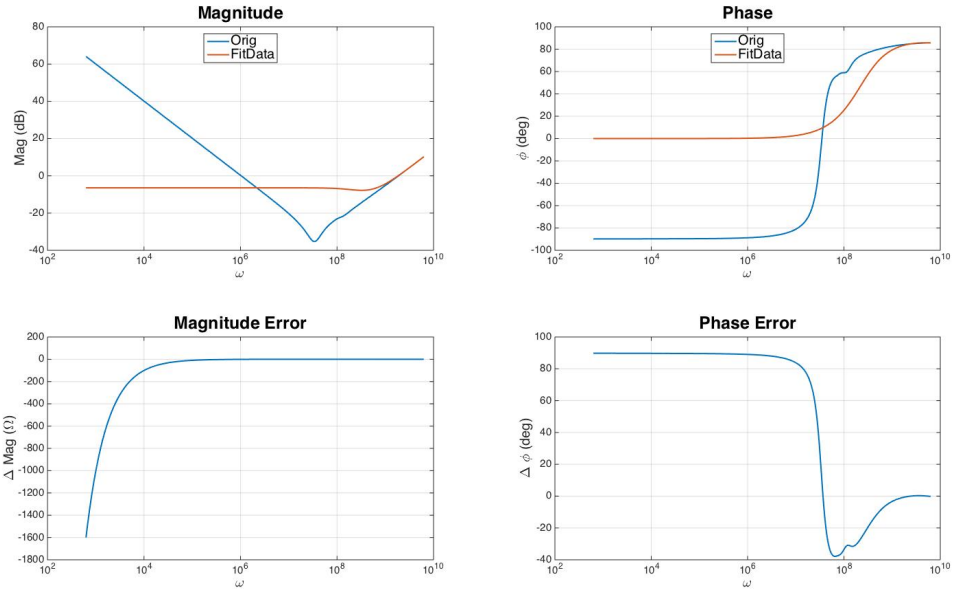


Figure 17: Levy's Technique

6.1.3 Weighted LSE

One improvement that can be made upon Levy's method is to iterate with a frequency dependent weighting function until the error term is minimized [35]. By multiplying Levy's error function by the weighting term in Equation: (37), we get Equation: (38), which can be minimized to obtain a new system of equations. The L subscript stands for the current iteration, while $L-1$ stands for the previous iteration.

$$W_{kL} = \frac{1}{|Q(jw_k)_{L-1}|^2} \text{ [35]} \tag{37}$$

$$E = \sum_{k=1}^{n} |\epsilon_k^{'}|^2 W_{kL} \text{ [35][Eq. 7]} \tag{38}$$

Equations (33), (34), (35), & (36) are the same, with Equations (29), (30), (31), & (32) being replaced with Equations: (39), (40), (41), & (42).

$$\lambda_h = \sum_{k=0}^{m} \omega_k^h W_{kL} \text{ [35][Eq. 9]} \tag{39}$$

$$S_h = \sum_{k=0}^{m} \omega_k^h R_k W_{kL} \text{ [35][Eq. 10]} \tag{40}$$

$$T_h = \sum_{k=0}^{m} \omega_k^h I_k W_{kL} \text{ [35][Eq. 11]} \tag{41}$$

$$U_h = \sum_{k=0}^{m} \omega_k^h (R_k^2 + I_k^2) W_{kL} \text{ [35][Eq. 12]} \tag{42}$$

Dependent upon the initial conditons and the model order, this particular method is not gauranteed to converge. Figure: 18 was generated with an initial condition of $Q(jw_k) == 1$ and a model order of 7. It shows that, for this data set, the squared error of the magnitude and phase do not converge after a particular number of iterations. Furthermore, they do not reach their minimums at the same iteration. In order to select the desired iteration, the magnitude and phase squared plots are normalized as in Equation: (43). The index of the minimum of Figure: 18 is selected as the best fit.

$$n = min(Emag/max(eMag) + Epha/max(ePha)) \tag{43}$$

Figure: 20 shows that this method can result in a much improved result over the Levy's original method, as seen in Figure: 17.

6.2 Modeling

This section will investigate several of the most common capacitor models. It will show how to fit them to a data set with Levy's method described in Section: 6.1, and

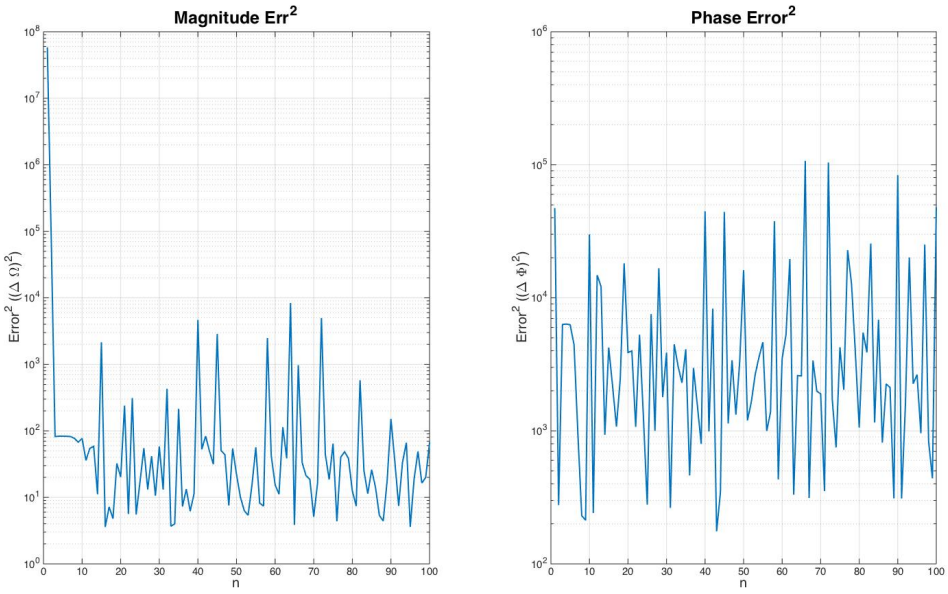


Figure 18: LSE + Iteration – Magnitude and Phase Error

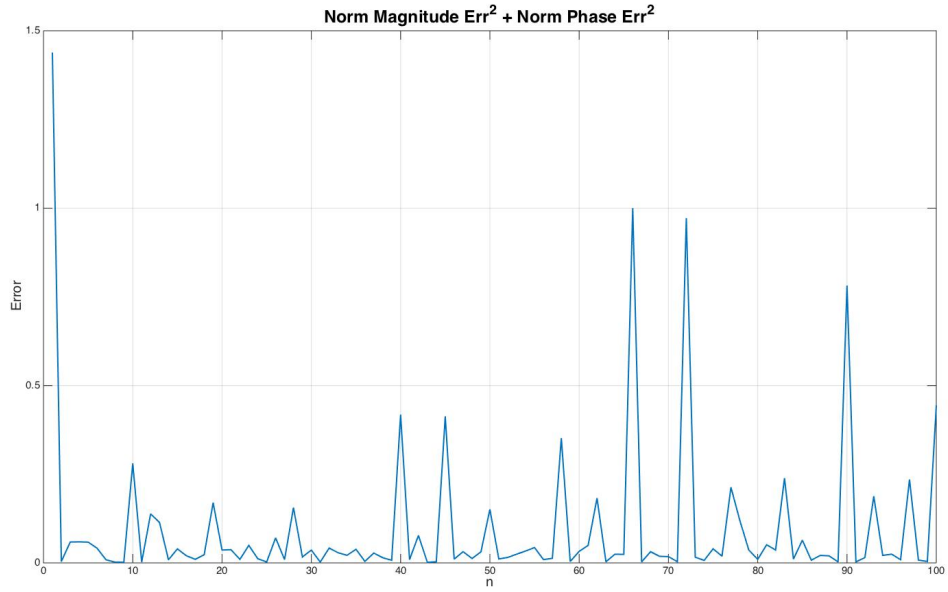


Figure 19: LSE + Iteration – Combined Error

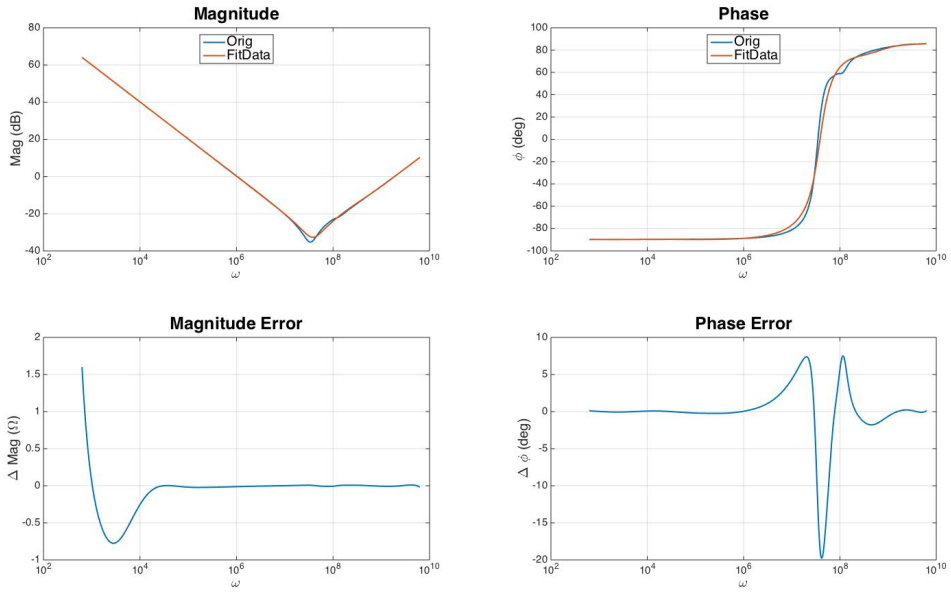


Figure 20: LSE + Iteration

will describe their effectiveness and limitations in doing so.

6.2.1 6 Term Model

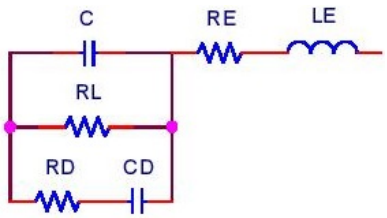


Figure 21: 6 Term Model

The model shown in Figure: 21 shows several of the most important parameters for a capacitor. It's impedance, shown in Equation: (44), and generalized in Equation:

(45) can be used as the basis for a regression analysis.

$$\begin{aligned}\bar{Z}(s) = &\frac{(R_E + R_L) + (L_E + C_D R_D R_E + C_D R_D R_L + C R_E R_L + C_D R_E R_L)s}{1 + (C_D R_D + C R_L + C_D R_L)s + C C_D R_D R_L s^2} \\ &+ \frac{(C_D L_E R_D + C L_E R_L + C_D L_E R_L + C C_D R_D R_E R_L)s^2 + C C_D L_E R_D R_L s^3}{1 + (C_D R_D + C R_L + C_D R_L)s + C C_D R_D R_L s^2}\end{aligned} \tag{44}$$

$$\bar{Z}(s) = \frac{a_0 + a_1 s + a_2 s^2 + a_3 s^3}{b_0 + b_1 s + b_2 s^2} \tag{45}$$

For this model, Equations: (34), (35), & (36) simplify down to Equations: (46), (47), & (48).

$$M = \begin{bmatrix} \lambda_0 & 0 & -\lambda_2 & 0 & T_1 & S_2 \\ 0 & \lambda_2 & 0 & -\lambda_4 & -S_2 & T_3 \\ \lambda_2 & 0 & -\lambda_4 & 0 & T_3 & S_4 \\ 0 & \lambda_4 & 0 & -\lambda_6 & -S_4 & T_5 \\ T_1 & -S_2 & -T_3 & S_4 & U_2 & 0 \\ S_2 & T_3 & -S_4 & -T_5 & 0 & U_4 \end{bmatrix} \tag{46}$$

$$N = \begin{bmatrix} A_0 \\ A_1 \\ A_2 \\ A_3 \\ B_1 \\ B_2 \end{bmatrix} \tag{47}$$

$$C = \begin{bmatrix} S_0 \\ T_1 \\ S_2 \\ T_3 \\ 0 \\ U_2 \end{bmatrix} \tag{48}$$

Using this model, the regression analysis method described in Section: 6.1.3 generates the output seen in Figure: 22. The fit tracks the original data well, except for at low frequencies and near resonance.

Substituting the polynomial coefficents seen in Equation: (56) into Equations: (49)-(55) returns the model parameter values as seen in Equation: (57). Even though the

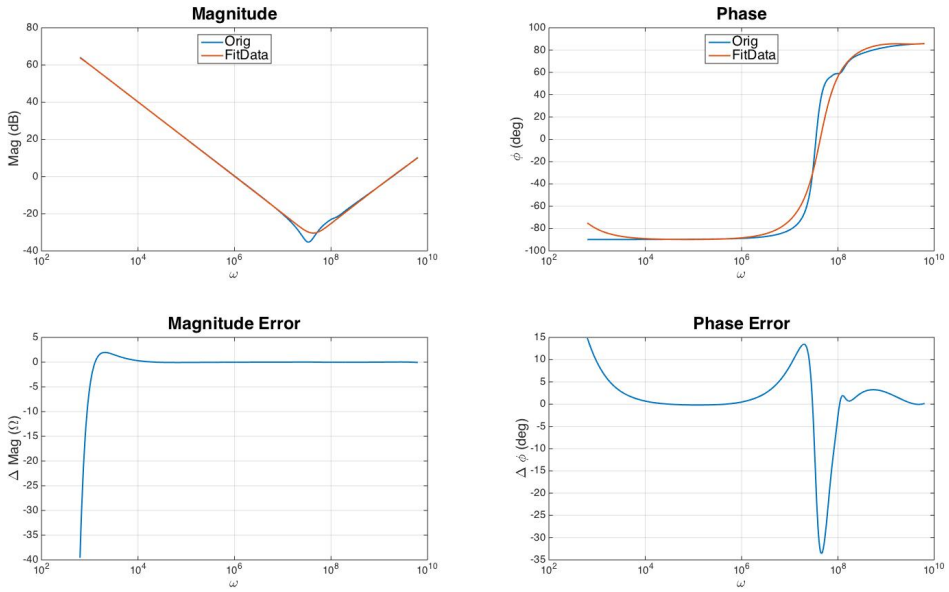


Figure 22: 6 Term Model: Bad Initilization

polynomial coefficients gave an acceptable fit to the data, it resulted in unacceptable circuit parameters. Since C and R_D are negative, this model is not physically realizable.

$$a_0 = R_E + R_L \tag{49}$$

$$a_1 = L_E + C_D R_D R_E + C_D R_D R_L + C R_E R_L + C_D R_E R_L \tag{50}$$

$$a_2 = C_D L_E R_D + C L_E R_L + C_D L_E R_L + C C_D R_D R_E R_L \tag{51}$$

$$a_3 = C C_D L_E R_D R_L \tag{52}$$

$$b_0 = 1 \tag{53}$$

$$b_1 = C_D R_D + C R_L + C_D R_L \tag{54}$$

$$b_2 = C C_D R_D R_L \tag{55}$$

$$
\begin{aligned}
a_0 &= 5.9991E^{+03} \\
a_1 &= 1.7934E^{-04} \\
a_2 &= 3.3158E^{-12} \\
a_3 &= 6.8295E^{-22} \\
b_0 &= 1.0000 \\
b_1 &= 5.9057E^{-03} \\
b_2 &= 1.4067E^{-12}
\end{aligned}
\qquad (56) \qquad
\begin{aligned}
C &= -8.2563E^{-10} \\
R_E &= 3.1886E^{-01} \\
L_E &= 4.8551E^{-10} \\
R_L &= 4.8551E^{-10} \\
C_D &= 9.8536E^{-07} \\
R_D &= -2.8824E^{-01}
\end{aligned}
\qquad (57)
$$

This problem stems from the author's [35] suggestion that the initial guess for $Q(jw_k) == 1$. While an initial guess is required to calculate the weighting function during the first iteration, this particular choice causes the problem seen in Equations: (58) & (59) for this application. They show that this initial condition does not start with rational solution for the circuit parameters.

$$
\begin{aligned}
a_0 &= 1 \\
a_1 &= 1 \\
a_2 &= 1 \\
a_3 &= 1 \\
b_0 &= 1 \\
b_1 &= 0 \\
b_2 &= 0
\end{aligned}
\qquad (58) \qquad
\begin{aligned}
C &= INF \\
R_E &= INF \\
L_E &= INF \\
R_L &= IND \\
C_D &= IND \\
R_D &= IND
\end{aligned}
\qquad (59)
$$

Starting instead with the rational set of circuit parameters seen in Equation: (61), the starting coefficients are as seen in Equation: (60).

$$
\begin{aligned}
a_0 &= 2 \\
a_1 &= 5 \\
a_2 &= 4 \\
a_3 &= 1 \\
b_0 &= 1 \\
b_1 &= 3 \\
b_2 &= 1
\end{aligned}
\tag{60}
$$

$$
\begin{aligned}
C &= 1 \\
R_E &= 1 \\
L_E &= 1 \\
R_L &= 1 \\
C_D &= 1 \\
R_D &= 1
\end{aligned}
\tag{61}
$$

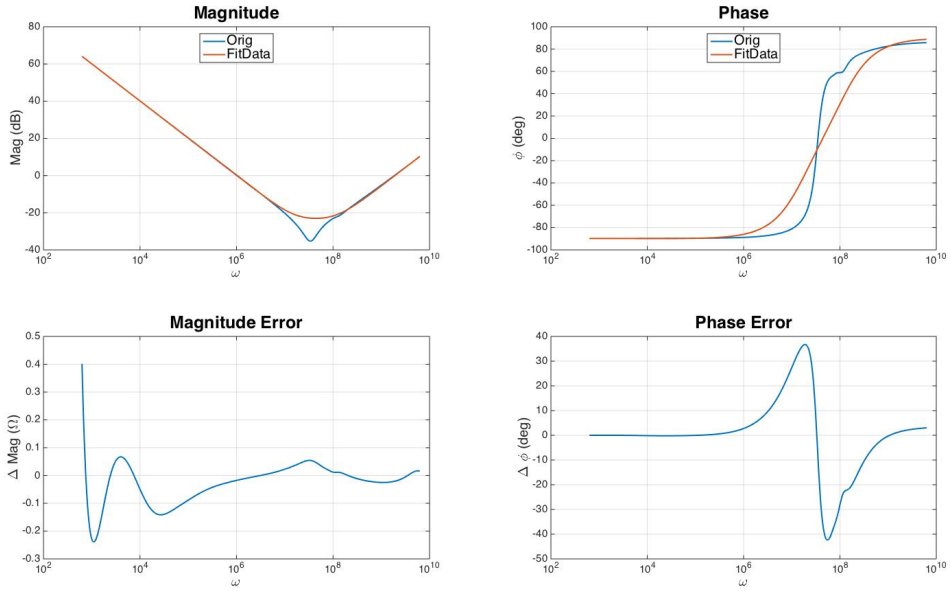


Figure 23: 6 Term Model: Good Initilization

Figure: 23 shows a good fit across the frequency spectrum for both magnitude and phase. The model does deviate at resonance, but that is unsurprising, as the number of parameters in the model is fairly low when compared with the model used to

generate the data.

7 Conclusion

This section will summarize the methodologies used to solve the stated problem.

8 Future Work

This section will describe the future work needed to be accomplished in order to complete and further the stated goals of this thesis. It will mostly focus on the practical circuitry implementation and other aspects needed to make it a viable tool.

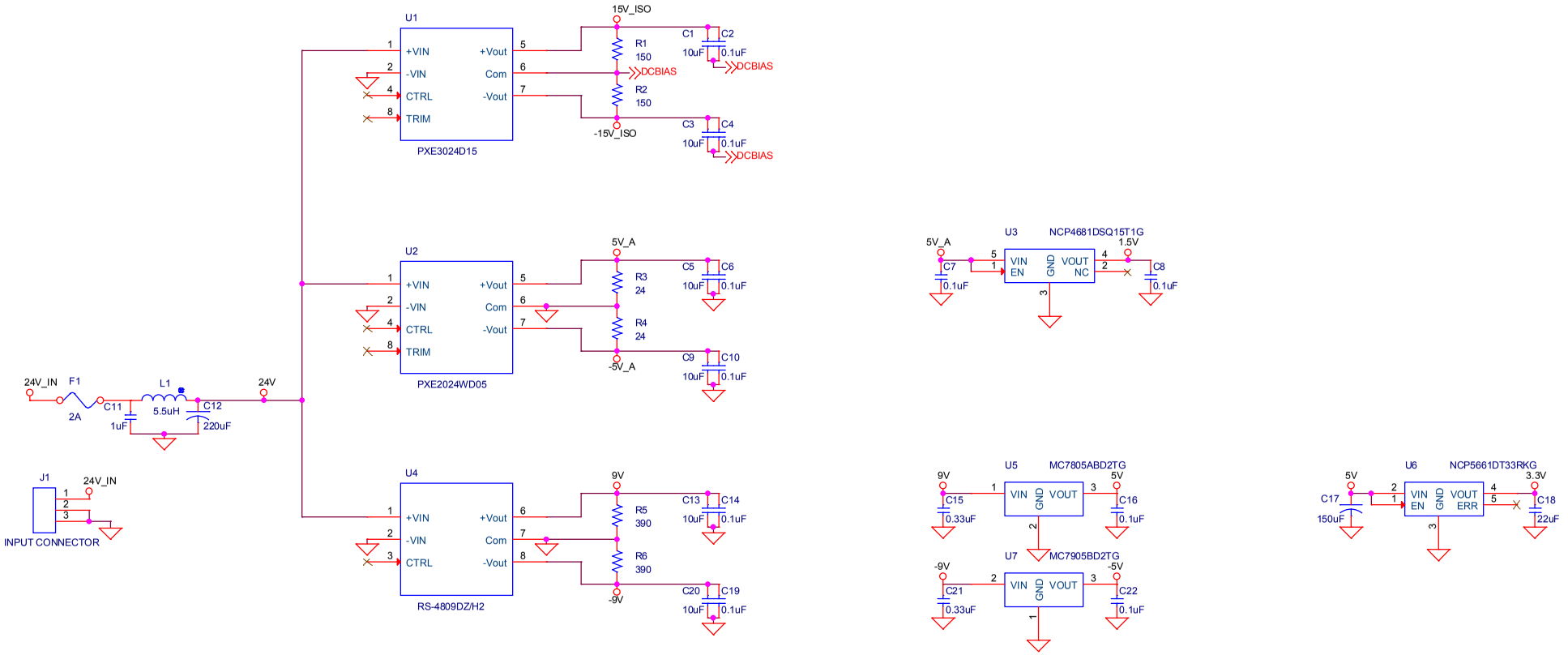
A Schematic

This section shows the schematic described in Section: 5.

TABLE OF CONTENTS

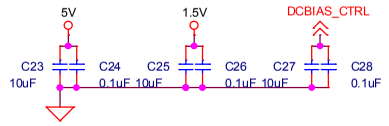
01 TABLE OF CONTENTS AND REVISION CONROL
02 POWER SUPPLIES
03 DC BIAS CONTROL
04 OPTOCOUPLER
05 DISCHARGE CIRCUITRY
06 FILTERING
07 IMPEDANCE / PHASE
08 ADC
09 MCU I/O
10 TEMPERATURE
11 COMMUNICATIONS

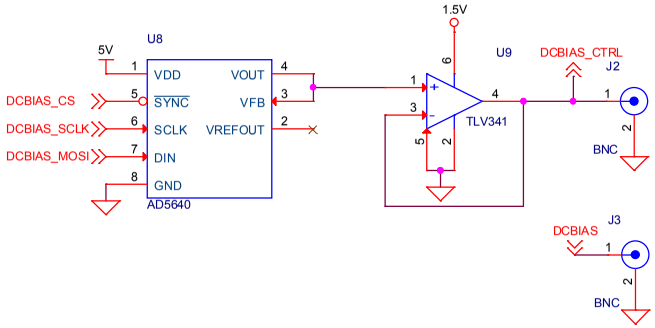
Title	TABLE OF CONTENTS	
Size B	Document Number DELIBERO_THESIS	Rev 1
Date: Thursday, June 25, 2015	Sheet 1 of 11	



POWER SUPPLIES

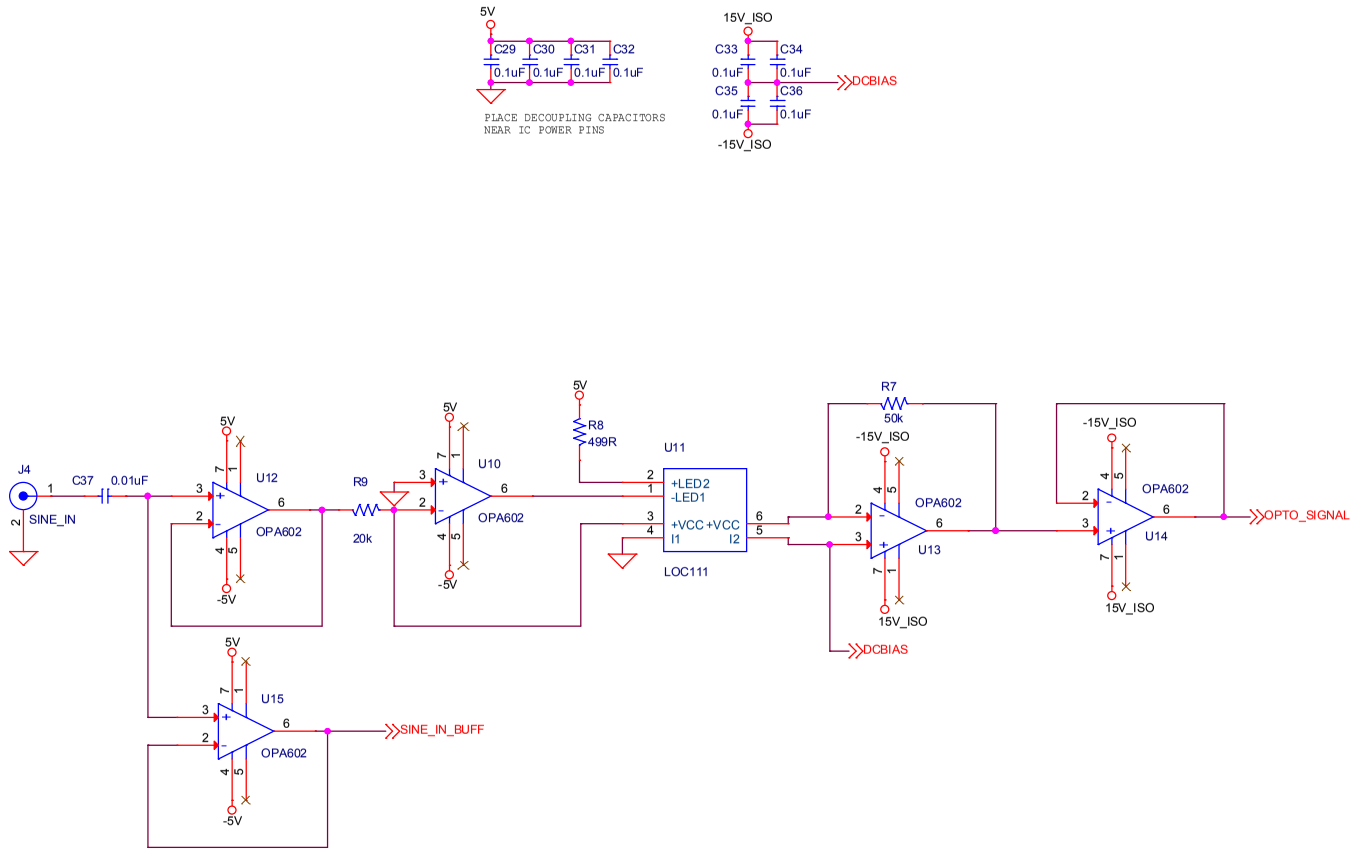
Title	POWER SUPPLIES	
Size B	Document Number DELIBERO_THESIS	Rev 1
Date: Wednesday, June 24, 2015	Sheet 2 of 11	





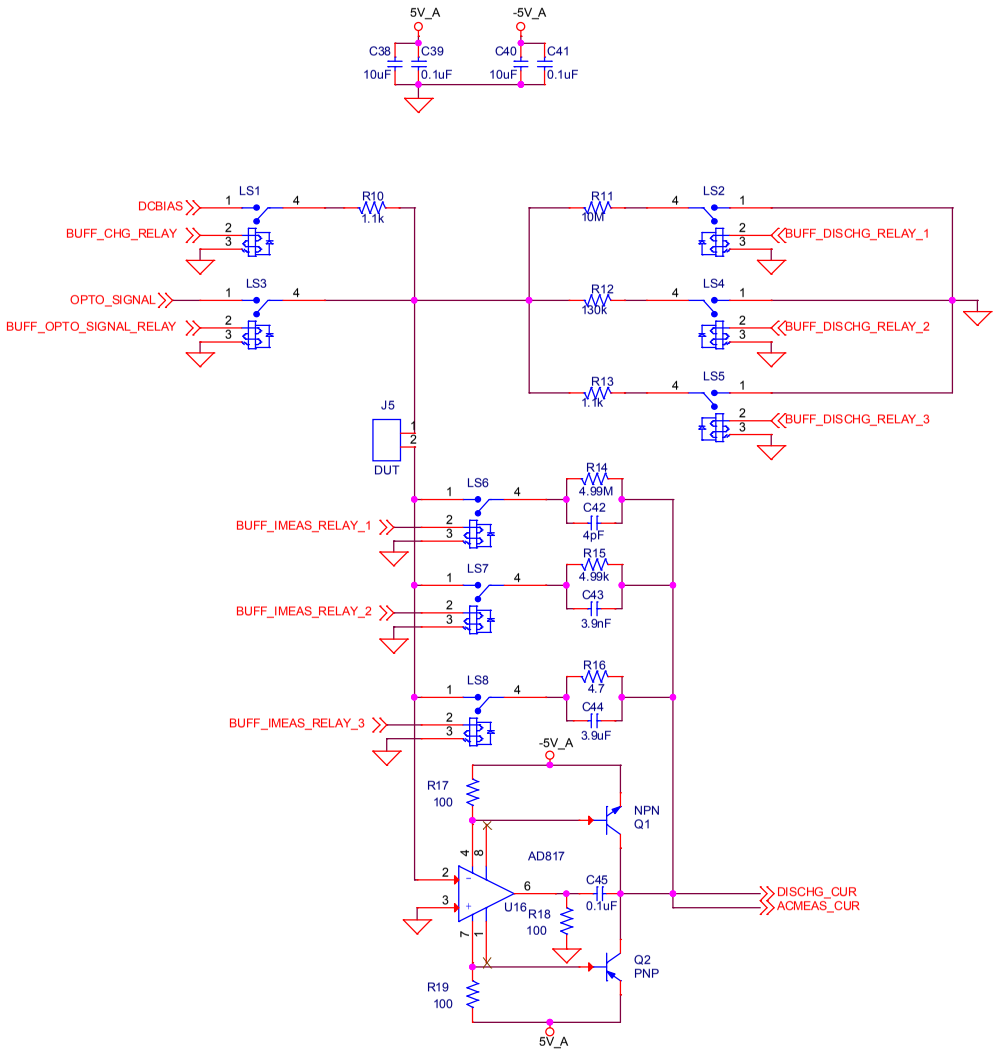
DC BIAS CONTROL

Title	DC BIAS CONTROL	
Size B	Document Number DELIBERO_THESIS	Rev 1
Date: Thursday, June 25, 2015	Sheet 3 of 11	



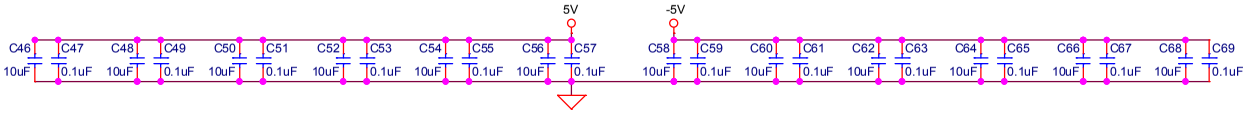
OPTOCOUPLER

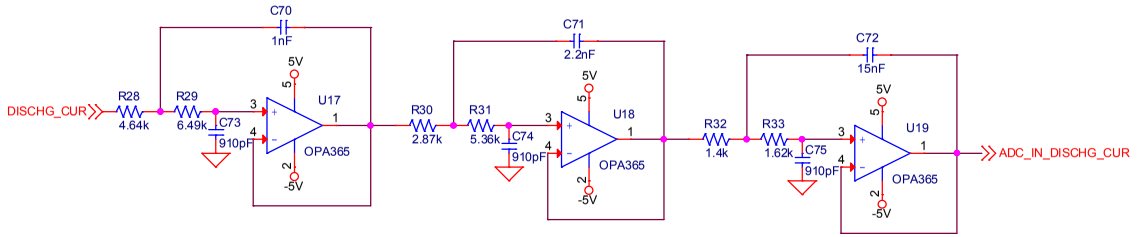
Title	OPTOCOUPLER	
Size B	Document Number DELIBERO_THESIS	Rev 1
Date: Thursday, June 25, 2015	Sheet 4 of 11	

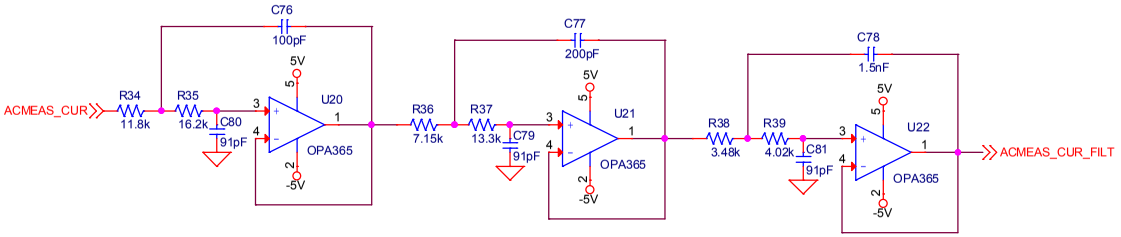


DISCHARGING CIRCUITRY

Title	DISCHARGE CIRCUITRY	
Size B	Document Number DELIBERO_THESIS	Rev 1
Date: Thursday, June 25, 2015	Sheet 5 of 11	

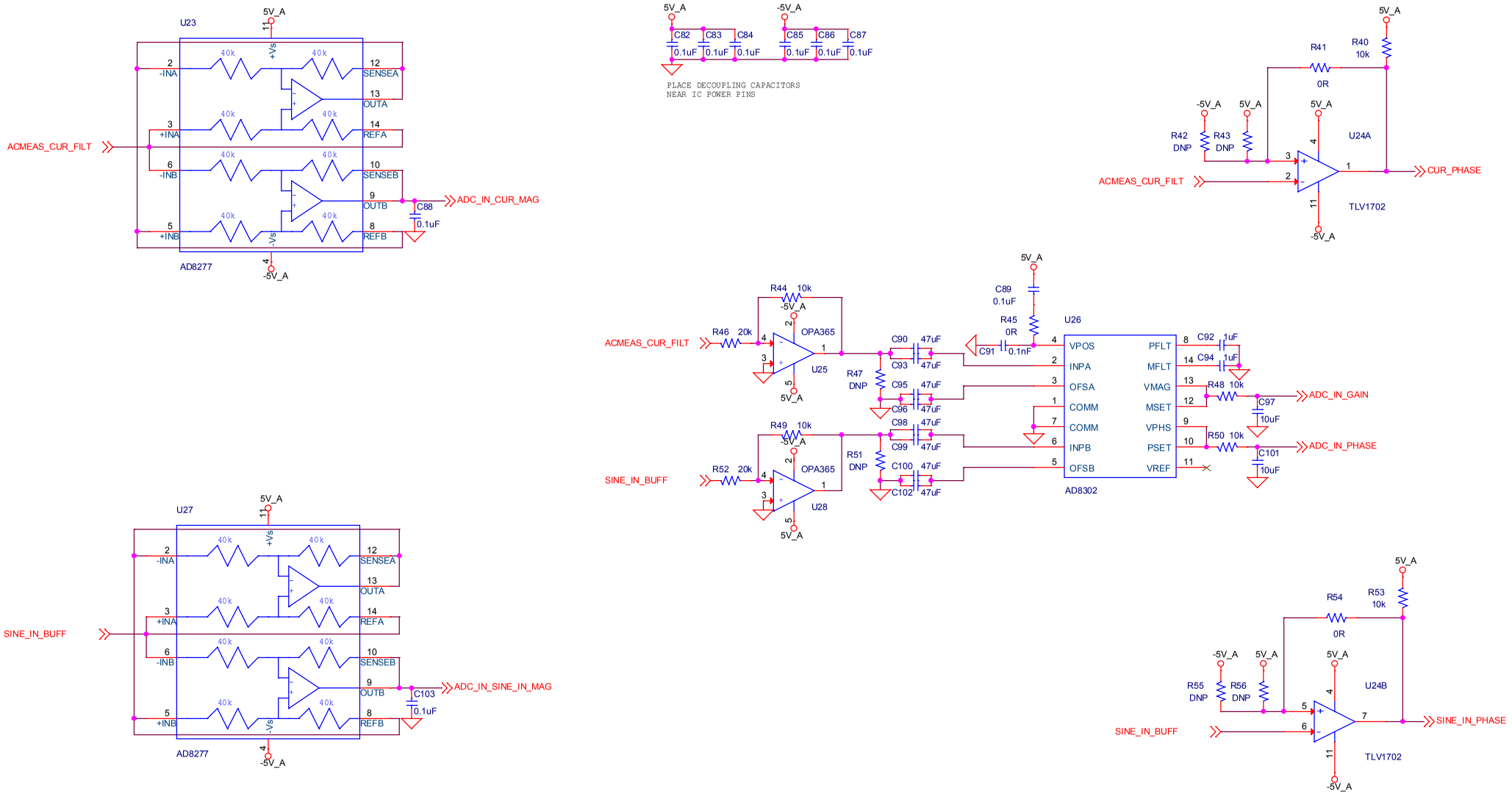






FILTERING

Title: FILTERING		
Size: B	Document Number: DELIBERO_THESIS	Rev: 1
Date: Thursday, June 25, 2015	Sheet 6 of 11	

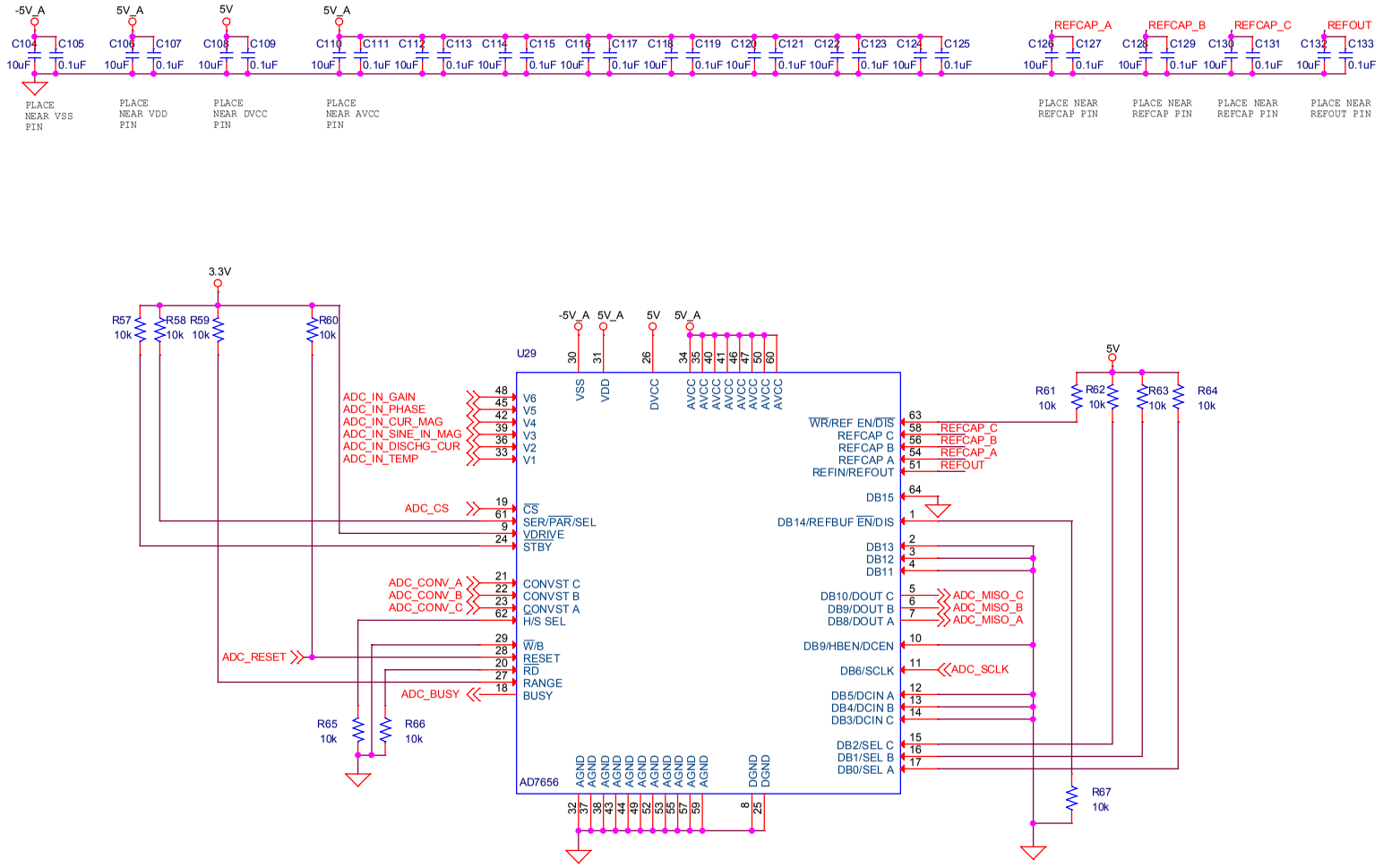


MAGNITUDE

PHASE

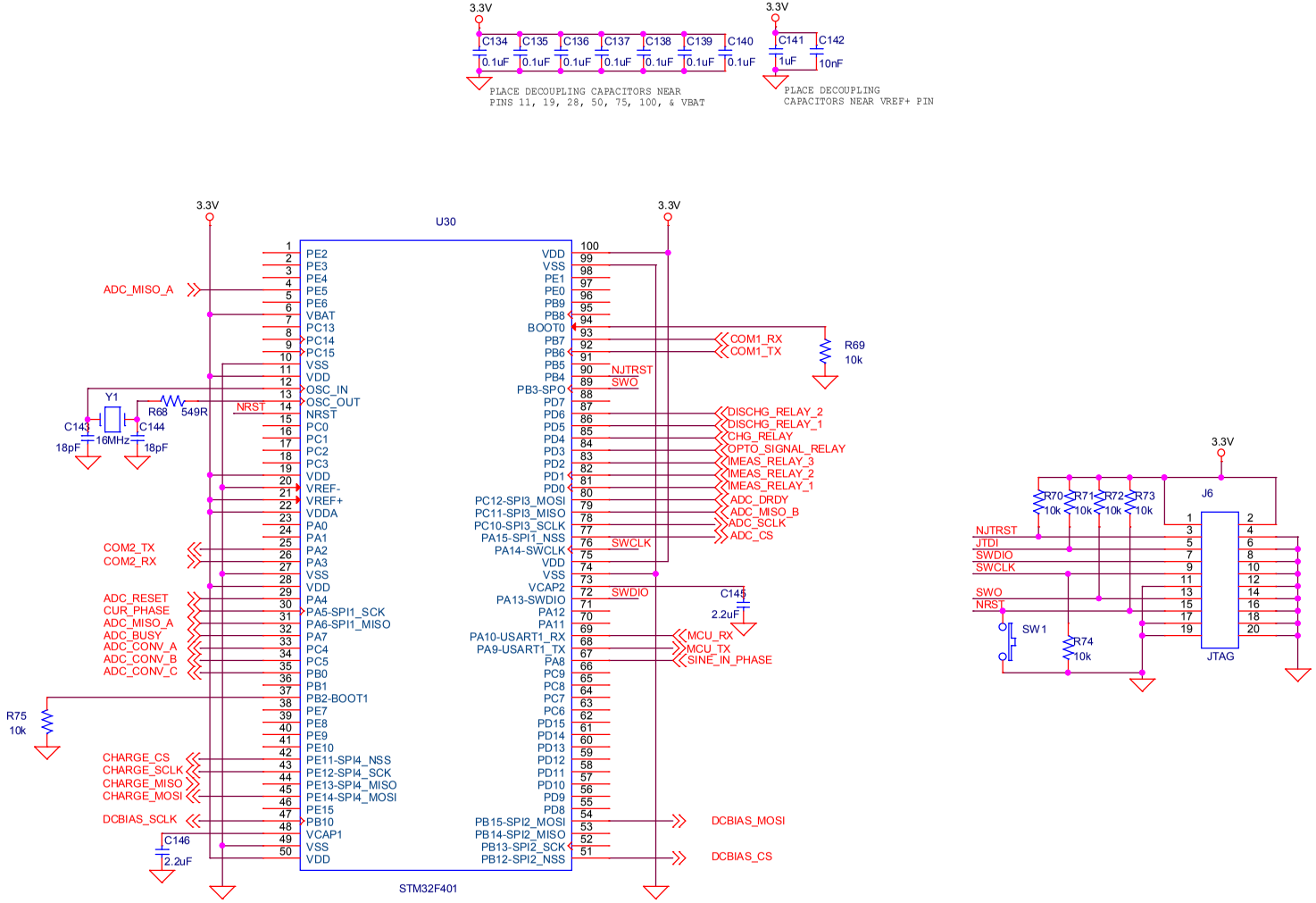
IMPEDANCE / PHASE

Title	IMPEDANCE / PHASE	
Size B	Document Number DELIBERO_THESIS	Rev 1
Date: Thursday, June 25, 2015	Sheet 7 of 11	



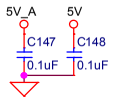
ADC

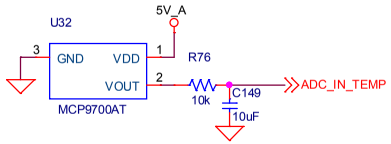
Title	ADC	
Size B	Document Number DELIBERO_THESIS	Rev 1
Date: Thursday, June 25, 2015	Sheet 8 of 11	



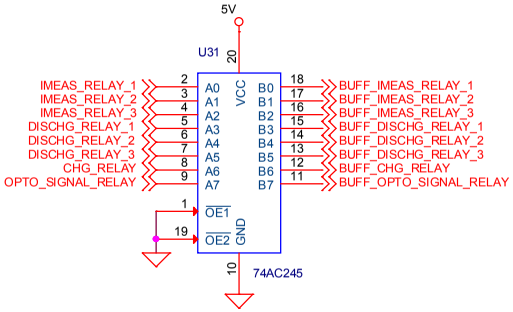
MCU I/O

Title	MCU I/O	
Size B	Document Number DELIBERO_THESIS	Rev 1
Date: Thursday, June 25, 2015	Sheet 9 of 11	





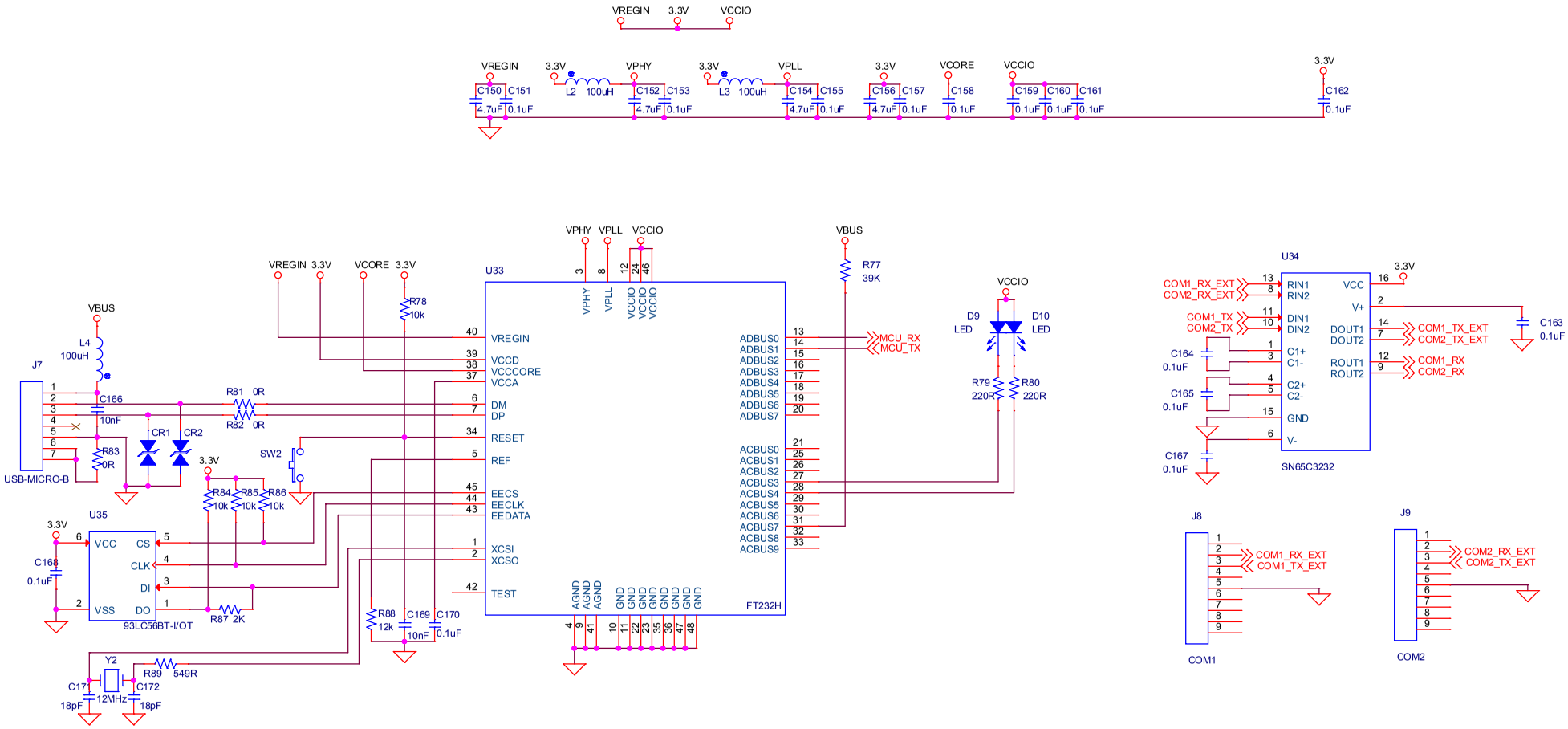
TEMPERATURE



BUFFER

MISCELLANEOUS

Title: MISCELLANEOUS		
Size: B	Document Number: DELIBERO_THESIS	Rev: 1
Date: Thursday, June 25, 2015	Sheet 10 of 11	



COMMUNICATIONS

Title	USB COMMUNICATIONS	
Size B	Document Number DELIBERO_THESIS	Rev 1
Date: Thursday, June 25, 2015	Sheet 11 of 11	

B Generating Modeling Images

This appendix will list all of the matlab code and supporting files needed to generate the images seen in Section: 6.

B.1 Example Data: Figure: 15

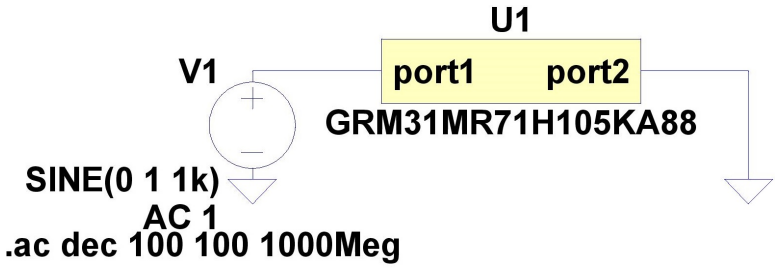


Figure 24: LTSpice Schematic for Capacitor Model

This section will describe how to obtain and generate the example data used for the regression fitting. This method uses LTSpice to generate impedance vs frequency data from one of Murata's capacitor models. First, go to Murata's online SimSurfing tool [36] and select the "Monolic Ceramic Capacitors" button. Download a SPICE *.mod file (Appendix: B.1.1) for the capacitor of interest, by selecting it from the list and clicking the "netlist" button. Open the *.mod file in LTSpice, right click on the part name in the line starting with ".SUBCKT," and select "Create Symbol." Create an LTSpice schematic similar to Figure: 24 and plot $\frac{V(n001)}{-I(V1)}$. The negative sign is important because LTSpice defines current as coming out of a node. If the negative sign is omitted, the phase will be offset by 180^o, and the regression analysis will solve for negative capacitance! With the plot window selected, select "$File \rightarrow Export \rightarrow Cartesian \rightarrow OK$" with the impedance plot selected as the waveform. Open the resultant *.txt file, delete the first line, and change all tabs to commas ("$<:> \%s/ < ctrl + v >< TAB > /,/g$" , "$\%s/\hat{}I/,/g$" in vim). Make sure to save the resultant file in "/scripts/data."

B.1.1 Capacitor Subcircuit Model

```
1  *---------------------------------------------------------------------
2  * SPICE Model generated by Murata Manufacturing Co., Ltd.
3  * Copyright(C) Murata Manufacturing Co., Ltd.
4  * Description :3216/X7R/1uF/50V
5  * Murata P/N :GRM31MR71H105KA88
6  * Property : C = 1[uF]
7  *---------------------------------------------------------------------
8  * Applicable Conditions:
9  *   Frequency Range = 100Hz-6000000000Hz
10 *   Temperature = 25 degC
11 *   DC Bias Voltage = 0V
12 *   Small Signal Operation
13 *---------------------------------------------------------------------
14 .SUBCKT GRM31MR71H105KA88 port1 port2
15 C1 port1 11 1.00e-6
16 L2 11 12 2.82e-11
17 R3 12 13 6.65e-3
18 C4 13 14 1.53e-4
19 R4 13 14 22.8
20 C5 14 15 2.02e-4
21 R5 14 15 1.85
22 C6 15 16 1.51e-4
23 R6 15 16 3.53e-1
24 C7 16 17 1.60e-4
25 R7 16 17 6.68e-2
26 C8 17 18 9.85e-5
27 R8 17 18 1.80e-2
28 C9 18 19 1.20e-4
29 R9 18 19 2.94e-3
```

```
L10 19 20 3.24e-11
R10 19 20 1.02e-1
L11 20 21 1.94e-10
R11 20 21 3.05e-2
L12 21 22 4.09e-11
R12 21 22 2.85e-2
C13 22 23 5.18e-6
L13 22 23 3.93e-11
R13 22 23 1.00e-2
C14 23 24 9.49e-7
L14 23 24 8.93e-11
R14 23 24 1.68e-2
C15 24 25 1.33e-11
L15 24 25 9.25e-11
R15 24 25 6.24
C16 25 port2 1.03e-12
L16 25 port2 3.80e-10
R16 25 port2 171
R100 port1 11 5.00e+8
.ENDS GRM31MR71H105KA88
```

B.1.2 Plot ExCapData Script

The functions used to get and plot the data can be found in Appendix: B.4.

```
% plot_ExCapData.m
% This script outputs a plot of the example data taken from ...
    Murata's Sim Surfing online tool.

% Clear environment
clearvars;
close all;
```

```
format shorte;

filename = './data/GRM31MR71H105KA88.txt';
[w, cData, rData, iData] = getData(filename);
myPlotType   = plotType.cData;
plotFit(myPlotType, cData, cData, w); % figures/modeling/levyIter.jpg
```

B.2 Basic LSE Image: Figure: 16

```
% run_basicLse.m
% This script runs the basic LSE for a line

% Clear environment
clearvars;
close all;
format shorte;

%% Generate data
n = 5;
err = rand(1,n)*1.3;
x = linspace(1,n,n);
b = rand(1)*ones(1,n);
y = x + err + b;

%% LSE
avg_y   = sum(y);
avg_x   = sum(x);
avg_xy  = sum(y .* x);
avg_xsq = sum(x .^ 2);

```

```
a_1 = (avg_xy - avg_x *avg_y) / (avg_xsq - avg_x^2)
a_0 = avg_y - a_1 * avg_x

y_lse = a_0*ones(1,length(x)) + a_1 .* x;

%% Plot
plotFit(plotType.DIFF_PLOT,y,y_lse,x);
```

B.3 Levy's Method: Figures: 17, 18, 19, & 20

```
% scripts/regression/run_levy_iter.m
% This script runs the example in Sanathanan's paper showing how
% to improve Levy's method by an iterative method.

% Clear environment
clearvars;
close all;
format shorte;

% Add funs to path
addpath('../utilityFuns');
addpath('../');

NumDeg = 7;
DenDeg = 7;
iterations = 100;
filename = '../data/GRM31MR71H105KA88.txt';

[w, cData, rData, iData] = getData(filename);
initDen = getInitGuess(w,modelTypes.NO_MODEL);
```

```
21 [G, numCoeffs, denCoeffs, E, minIndex] = ...
22 regression_levy_iter(cData, w, iterations, NumDeg, DenDeg, initDen);
23
24 %% Error minimization
25 Emag = E(:,1);
26 Epha = E(:,2);
27 EmagNorm = Emag ./ max(Emag);
28 EphaNorm = Epha ./ max(Epha);
29 E2 = EmagNorm + EphaNorm;
30 n = find(E2==min(E2),1);
31 fprintf('Error Minimized at Iteration: %i\n',n);
32
33 %% Plot
34 plotcDiff = plotType.cVectorsDiff;
35 plotErrs  = plotType.twoErrors;
36 plotErr   = plotType.oneError;
37
38 plotFit(plotcDiff, cData, G(n,:), w); % figures/modeling/levyIter.jpg
39 plotFit(plotErrs , Emag , Epha); % figures/modeling/levyIter_Err1.jpg
40 plotFit(plotErr  , E2         ); % figures/modeling/levyIter_Err2.jpg
```

```
1  % scripts/getInitGuess.m
2  % This function returns the appropriate initial guess
3  % for Sanathanan's extension to Levy's method.
4  function [Den] = getInitGuess(w,modelType)
5      if modelType == modelTypes.FULL_MODEL
6          C  = 1;
7          RE = 1;
8          LE = 1;
9          RL = 1;
10         CD = 1;
```

```
11          RD = 1;
12
13          b1 = CD*RD+C*RL+CD*RL;
14          b2 = C*CD*RD*RL;
15          [¬,¬,Den] = calcDataFromCoeffs([1;1;1],[1,b1,b2],w);
16
17      else % Set denominator to all ones
18          Den = ones(length(w));
19      end % if modelType == modelTypes.FULL_MODEL
20  end % function [Den] = getInitGuess(w)
```

```
1  % scripts/regression_levy_iter.m
2  % This script runs the calculations for iterating Levy's method
3  function [G, numCoeffs, denCoeffs, E, minIndex] = ...
       regression_levy_iter(cData, w, iterations, numbNumCoeffs, ...
       numbDenCoeffs, initDen)
4      % G(s) = \frac{a0 + a1*s + a2*s^2 + ... + an*s^n}
5      %             {b0 + b1*s + b2*s^2 + ... + bm*s^m}
6
7      % Solve
8      numCoeffs = ones(iterations,numbNumCoeffs);
9      denCoeffs = ones(iterations,numbDenCoeffs);
10     W         = ones(length(w)); % Weighting function
11
12     % Set initial guess of the denomiator. This also will cause it
13     % to calc Levy's original method if the iteraiton == 1.
14     Den = initDen;
15     G = zeros(iterations,length(w));
16
17     for iter = 1:iterations
18         W = 1 ./ abs(Den).^2;
```

```
19          [numCoeffs(iter,:), denCoeffs(iter,:)] = ...
                calcCoeffs(cData, w, W, numbNumCoeffs, numbDenCoeffs);
20          [G(iter,1:length(w)),¬,Den] = ...
                calcDataFromCoeffs(numCoeffs(iter,:),denCoeffs(iter,:),w);
21          E(iter,:) = sumError(cData,G(iter,1:length(w)));
22          iter
23      end
24
25      Emag = E(:,1);
26      Epha = E(:,2);
27      EmagNorm = Emag ./ max(Emag);
28      EphaNorm = Epha ./ max(Epha);
29      E2 = EmagNorm + EphaNorm;
30      minIndex = find(E2==min(E2),1);
31  end % function [G,numCoeffs,denCoeffs] = regression_levy_iter...
32
33  % This calculates the polynomial coeffs from the polynomial series
34  function [numCoeffs, denCoeffs] = calcCoeffs(cData, w, W, ...
        numbNumCoeffs, numbDenCoeffs)
35
36      % Prepare variables
37      Mrows = numbNumCoeffs + numbDenCoeffs - 1;
38      Mcols = Mrows;
39
40      M = ones(Mrows,Mcols);
41      N = ones(Mrows,1);
42      C = ones(Mrows,1);
43
44      sign  = 0;
45      power = 0;
46
47      %% Calculate
```

```
48      % Populate M
49      for row = 1:Mrows
50          for col = 1:Mcols
51              % Find out which section of M you are in:
52              if ((row ≤ numbNumCoeffs) && (col ≤ numbNumCoeffs))
53                  %Upper Left Quadrant -- lambdas
54                  xrow = row;
55                  xcol = col;
56
57                  if (xor(mod(xrow,2) == 0, mod(xcol,2) == 0) == 1)
58                  % If both are odd or both are even
59                      M(xrow,xcol) = 0;
60
61                  else
62                      power = xrow + xcol - 2;
63
64                      if ((mod(xcol,3) == 0) && (mod(xrow,2) == 1))
65                          sign = -1;
66                      elseif ((mod(xcol,4) == 0) && (mod(xrow,2) == 0))
67                          sign = -1;
68                      else
69                          sign =  1;
70                      end
71
72                      M(xrow,xcol) = sign * lambda(W,w,power);
73                  end % else
74
75
76              elseif (row ≤ numbNumCoeffs)
77                  % Upper Right quadrant -- S and T
78                  xrow = row;
79                  xcol = col - numbNumCoeffs;
```

```
80                  power = xrow + xcol - 1;
81
82                  if      (mod(xcol,4) == 1) && (mod(xrow,2) == 1)
83                      sign =  1;
84                  elseif (mod(xcol,4) == 1) && (mod(xrow,2) == 0)
85                      sign = -1;
86                  elseif (mod(xcol,4) == 2)
87                      sign =  1;
88                  elseif (mod(xcol,4) == 3) && (mod(xrow,2) == 1)
89                      sign = -1;
90                  elseif (mod(xcol,4) == 3) && (mod(xrow,2) == 0)
91                      sign = 1;
92                  elseif (mod(xcol,4) == 4)
93                      sign = -1;
94                  end
95
96                  if  (mod(power,2) == 1)
97                      M(row,col) = sign * T(W,w,imag(cData),power);
98                  else
99                      M(row,col) = sign * S(W,w,real(cData),power);
100                 end
101
102
103             elseif ((row > numbNumCoeffs) && (col ≤ numbNumCoeffs))
104                 % Lower Left -- S and T
105                 xrow = row - numbNumCoeffs;
106                 xcol = col;
107                 power = xrow + col - 1;
108
109                 if      (mod(xcol,4) == 1)
110                     sign =  1;
111                 elseif (mod(xcol,4) == 2) && (mod(xrow,2) == 1)
```

```
112                     sign = -1;
113                 elseif (mod(xcol,4) == 2) && (mod(xrow,2) == 0)
114                     sign =  1;
115                 elseif (mod(xcol,4) == 3)
116                     sign = -1;
117                 elseif (mod(xrow,2) == 1)
118                     sign =  1;
119                 elseif (mod(xrow,2) == 0)
120                     sign = -1;
121                 end
122
123                 if (mod(power,2) == 0)
124                     M(row,col) = sign * S(W,w,real(cData),power);
125                 else
126                     M(row,col) = sign * T(W,w,imag(cData),power);
127                 end
128
129             else
130                 % Lower Right -- U
131                 xrow = row - numbNumCoeffs;
132                 xcol = col - numbNumCoeffs;
133
134                 % If both are odd or both are even
135                 if (xor(mod(xrow,2) == 0, mod(xcol,2) == 0) == 1)
136                     M(row,col) = 0;
137                 else
138                     power = xrow + xcol;
139
140                     if     (mod(xcol,4) == 1)
141                         sign = 1;
142                     elseif (mod(xcol,4) == 2)
143                         sign = 1;
```

```
144                     else
145                         sign = -1;
146                     end
147
148                     M(row,col) = sign * ...
                            U(W,w,real(cData),imag(cData),power);
149                 end % else -- check for zeros
150             end % else -- searching through quadrants
151         end % for row = 1:Mrows
152     end % for col = 1:Mcols
153
154     % Populate C
155     for (row = 1:length(C))
156         if (row ≤ numbNumCoeffs)
157             xrow = row;
158             power = xrow - 1;
159             if (mod(power,2) == 0)
160                 C(xrow) = S(W,w,real(cData),power);
161             else
162                 C(xrow) = T(W,w,imag(cData),power);
163             end
164         else
165             xrow = row - numbNumCoeffs;
166             power = xrow;
167             if (mod(xrow,2) == 1)
168                 C(row) = 0;
169             else
170                 C(row) = U(W,w,real(cData),imag(cData),power);
171             end
172         end
173     end
174
```

```
175     % Solve For N
176     N = inv(M) * C;
177
178     numCoeffs = N(1:numbNumCoeffs);
179     denCoeffs = [1;N(numbNumCoeffs+1:length(N))]; % Add 1 as coeff_0
180
181     %Sub Functions
182     function [Lh] = lambda(W,wk,h)
183         Lh = 0;
184         for k = 1:1:length(wk)
185             Lh = Lh + wk(k)^h*W(k);
186         end
187     end
188
189     function [S] = S(W,wk,Rk,h)
190         S = 0;
191         for k = 1:1:length(wk)
192             S = S + wk(k)^h * Rk(k)*W(k);
193         end
194     end
195
196     function [T] = T(W,wk,Ik,h)
197         T = 0;
198         for k = 1:1:length(wk)
199             T = T + wk(k)^h * Ik(k)*W(k);
200         end
201     end
202
203     function [U] = U(W,wk,Rk,Ik,h)
204         U = 0;
205         for k = 1:1:length(wk)
206             U = U + wk(k)^h * (Rk(k)^2 + Ik(k)^2)*W(k);
```

```
207         end
208     end
209 end % function [b,a] = calcCoeffs(w_norm, T, cData)
```

B.4 Utility Functions

This appendix holds common "utility" functions used by many of the MATLAB scripts. You are required to manually save each plot after calling the "plotfit.m."

```
1  % Getdata.m
2  % This function parses data from an LTSpice file into a
3  % format that can be used for regression analysis.
4  function [w, cData, rData, iData] = getData(filename)
5      data  = csvread(filename);
6      w     = (data(:,1) * 2 * pi)';
7      rData = data(:,2)';
8      iData = data(:,3)';
9      cData = rData + 1i * iData;
10 end
```

```
1  % plotFit.m
2  % Outputs different plots for the regression analysis.
3  function [h_fig, h_leg, h_title, ax, p] = plotFit(pType, Data1, ...
       Data2, x)
4  %% Common Prep
5  titleSize     = 25;
6  legendSize    = 20;
7  axisTitleSize = 20;
8  axisSize      = 15;
9
```

```
10 h_fig = figure;
11
12 %% Plot specific request
13 if pType == plotType.DIFF_PLOT;
14     ax = subplot(1,1,1);
15     p(1) = plot(x,Data1,'o'); hold on;
16     p(2) = plot(x,Data2);
17     h_title = title('Basic LSE','FontSize',titleSize);
18     h_leg = legend('Orig','Fit');
19     set(h_leg,'FontSize',legendSize);
20     xlabel(ax, 'x', 'FontSize', axisTitleSize);
21     ylabel(ax, 'y', 'FontSize', axisTitleSize);
22
23 elseif pType == plotType.cData
24     [ax,p(1),p(2)] = plotyy(x,mag2db(abs(Data1)),x, ...
25     rad2deg(phase(Data1)),'semilogx','semilogx');
26
27     h_title = title('GRM31MR71H105KA88','FontSize',titleSize);
28     h_leg = legend('Magnitude','Phase','Location','north');
29     set(h_leg,'FontSize',legendSize);
30     xlabel(ax(1), '\omega (rad)'  , 'FontSize', axisTitleSize);
31     ylabel(ax(1), 'Mag (dB)'      , 'FontSize', axisTitleSize);
32     ylabel(ax(2), '\phi (deg)'    , 'FontSize', axisTitleSize);
33     min_y1 = min(get(ax(1),'ytick'));
34     max_y1 = max(get(ax(1),'ytick'));
35     min_y2 = min(get(ax(2),'ytick'));
36     max_y2 = max(get(ax(2),'ytick'));
37     set(ax(1),'ytick',min_y1:20:max_y1);
38     set(ax(2),'ytick',min_y2:20:max_y2);
39
40 elseif pType == plotType.MULTCDATA
41     colors = [0.000,0.447,0.741 ;
```

```
42      0.850,0.325,0.098 ;
43      0.929,0.694,0.125 ;
44      0.494,0.184,0.556 ;
45      0.466,0.674,0.188 ;
46      0.301,0.745,0.933];
47
48      [ax(1,:),p(1),p(2)] = plotyy(x,mag2db(abs(Data1(1,:))),x,...
49      rad2deg(phase(Data1(1,:))),'semilogx','semilogx'); hold on;
50      [ax(2,:),p(3),p(4)] = plotyy(x,mag2db(abs(Data1(2,:))),x,...
51      rad2deg(phase(Data1(2,:))),'semilogx','semilogx'); hold on;
52      [ax(3,:),p(5),p(6)] = plotyy(x,mag2db(abs(Data1(3,:))),x,...
53      rad2deg(phase(Data1(3,:))),'semilogx','semilogx');
54
55      h_title = title('RC Filter -- Increasing ...
            C','FontSize',titleSize);
56      h_leg = legend(p,'Mag1','Pha1','Mag2','Pha2',...
57      'Mag3','Pha3','Location','northeast');
58      set(h_leg,'FontSize',legendSize);
59      xlabel(ax(1,1), '\omega (rad)'  , 'FontSize', axisTitleSize);
60      ylabel(ax(1,1), 'Mag (dB)'      , 'FontSize', axisTitleSize);
61      ylabel(ax(1,2), '\phi (deg)'    , 'FontSize', axisTitleSize);
62
63      for ind = 1:6
64          p(ind).Color = colors(ind,:);
65          ax(ind).YColor = [0,0,0]; % Black
66      end
67
68      p(2).Marker = 'o';
69      p(4).Marker = 'o';
70      p(6).Marker = 'o';
71
72  elseif pType == plotType.cVectorsDiff
```

```
73      rows = 2;
74      cols = 2;
75
76      ax(1) = subplot(rows,cols,1);
77      p(1)  = semilogx(x,mag2db(abs(Data1))); hold on;
78      p(2)  = semilogx(x,mag2db(abs(Data2)));
79      h_title(1) = title('Magnitude','FontSize',titleSize);
80      h_leg = legend('Orig','FitData','Location','north');
81      set(h_leg,'FontSize',legendSize);
82      xlabel('\omega'  ,'FontSize',axisTitleSize);
83      ylabel('Mag (dB)','FontSize',axisTitleSize);
84      min_y1 = min(get(ax(1),'ytick'));
85      max_y1 = max(get(ax(1),'ytick'));
86      set(ax(1),'ytick',min_y1:20:max_y1);
87
88      ax(2) = subplot(rows,cols,2);
89      p(3)  = semilogx(x,rad2deg(phase(Data1))); hold on;
90      p(4)  = semilogx(x,rad2deg(phase(Data2)));
91      h_title(2) = title('Phase','FontSize',titleSize);
92      h_leg = legend('Orig','FitData','Location','north');
93      set(h_leg,'FontSize',legendSize);
94      xlabel('\omega','FontSize',axisTitleSize);
95      ylabel('\phi (deg)','FontSize',axisTitleSize);
96      min_y2 = min(get(ax(2),'ytick'));
97      max_y2 = max(get(ax(2),'ytick'));
98      set(ax(2),'ytick',min_y2:20:max_y2);
99
100     ax(3) = subplot(rows,cols,3);
101     p(5)  = semilogx(x,abs(Data2)-abs(Data1));
102     h_title(3) = title('Magnitude Error','FontSize',titleSize);
103     xlabel('\omega','FontSize',axisTitleSize);
104     ylabel('\Delta Mag (\Omega)','FontSize',axisTitleSize);
```

```
105
106     ax(4) = subplot(rows,cols,4);
107     p(6)  = semilogx(x,rad2deg(phase(Data2))-rad2deg(phase(Data1)));
108     h_title(4) = title('Phase Error','FontSize',titleSize);
109     xlabel('\omega','FontSize',axisTitleSize);
110     ylabel('\Delta \phi (deg)','FontSize',axisTitleSize);
111
112 elseif pType == plotType.oneError
113     rows = 1;
114     cols = 1;
115     ax = subplot(rows,cols,1);
116     p  = plot(Data1);
117     h_title = title('Norm Magnitude Err^2 + Norm Phase ...
            Err^2','FontSize',titleSize);
118     h_leg = legend();
119     xlabel('n','FontSize',axisTitleSize);
120     ylabel('Error','FontSize',axisTitleSize);
121
122 elseif pType == plotType.twoErrors
123     rows = 1;
124     cols = 2;
125
126     ax(1) = subplot(rows,cols,1);
127     p (1) = semilogy(Data1);
128     h_title(1) = title('Magnitude Err^2','FontSize',titleSize);
129     xlabel('n','FontSize',axisTitleSize);
130     ylabel('Error^2 ((\Delta \Omega)^2)','FontSize',axisTitleSize);
131
132     ax(2) = subplot(rows,cols,2);
133     p (2) = semilogy(Data2);
134     h_title(2) = title('Phase Error^2','FontSize',titleSize);
135     xlabel('n','FontSize',axisTitleSize);
```

```
136     ylabel('Error^2 ((\Delta \Phi)^2)','FontSize',axisTitleSize);
137 end % if pType == plotType.cVectorsDiff
138 
139 %% Common plotting options
140 for ind = 1:length(ax)
141         grid(ax(ind),'on');
142         set(ax(ind),'FontSize',axisSize);
143     end % for ind = 1:length(ax)
144 
145     for ind = 1:length(p)
146         p(ind).LineWidth = 2;
147     end % for ind = 1:length(p)
148 end % function plotFit()
```

```
1 % plotType.m
2 % Holds the enum for the various plot types.
3 classdef plotType
4     enumeration
5         DIFF_PLOT, MULTCDATA, cData, cVectorsDiff, oneError, ...
              twoErrors
6     end
7 end
```

References

[1] Avx corporation history.

[2] Relva C. Buchanan, editor. *Ceramic Materials for Electronics*. Marcel Dekker, 3 edition, June 2004.

[3] What are mica capacitors?, 2014.

[4] PETER CORWIN. *Synthesis and Characterization of Titanium Zirconium Based Alloys for Capacitor Use*. PhD thesis, Case Western Reserve University, 2013.

[5] G W A Dummer. *Electronics Inventions and Discoveries*. Institute of Physics Publishing, 1997.

[6] Steven Ehret. *Instrumentation For Anodization and In-Situ Testing of Titanium Alloys for Capacitor Anodes*. Masters thesis, Case Western Reserve University, January 2012.

[7] JD Fast. The transition point diagram of the zirconium-titanium system. *Recueil des Travaux Chimiques des Pays-Bas*, 58(11):973–983, 1939.

[8] Richard Fore. Understanding temperature coefficients of ceramics, January 2005.

[9] S. Fujishima. The history of ceramic filters. *Ultrasonics, Ferroelectrics and Frequency Control, IEEE Transactions on*, 47(1):1–7, Jan 2000.

[10] James M. Gleason. Steatite for high frequency insuation. In *Journal of the British Institution of Radio Engineers*. Institute of Radio Engineers, 1945.

[11] J. Ho, T.R. Jow, and S. Boggs. Historical introduction to capacitor technology. *Electrical Insulation Magazine, IEEE*, 26(1):20–25, January 2010.

[12] Brian Holman. *The Electrical Characterization of Tantalum Capacitors as MIS Devices*. ProQuest LLC, 2008.

[13] Capacitors.

[14] B. Jaffe, W.R Cook Jr., and H. Jaffe. *Piezoelectric Ceramics*. Adacemic Press Ince (London) Ltd, 1971.

[15] J.M.Herbert. *Ceramic Dielectrics in Capacitors*. Gordon and Breach Scientific Publishers, 1985.

[16] Jill Jonnes. *Empires of Light*. Random House, 2003.

[17] James Karki. Analysis of the sallen-key architecture. Online, September 2002.

[18] Jun-Wan Ki. *Titanium sponge on titanium substrate for titanium electrolytic capacitor anodes*. PhD thesis, Case Western Reserve University, 2005.

[19] Equo Kobayashi, Shigeru Matsumoto, Takayuki Yoneyama, Hitoshi Hamanaka, et al. Mechanical properties of the binary titanium-zirconium alloys and their potential for biomedical materials. *Journal of biomedical materials research*, 29(8):943–950, 1995.

[20] Lambda. Pxe series (dual output) dc-dc converters, July 2005.

[21] Learning about electronics.

[22] E. C. Levy. Complex-curve fitting. *Automatic Control, IRE Transactions on*, AC-4(1):37–44, 1959.

[23] Loc series linear optocouplers application note: An-107, September 2013.

[24] An efficiency primer for switch-mode, dcdc converter power supplies, December 2008.

[25] Francis Merat Michael DeLibero, Steven Ehret. Instrumentation for the anodization and characterization of titanium electrodes for electrolytic capacitors. In *Energytech, 2012 IEEE*. IEEE, May 2012.

[26] J.R. Miller. Introduction to electrochemical capacitor technology. *Electrical Insulation Magazine, IEEE*, 26(4):40–47, July 2010.

[27] Reem Malik Moshe Gerstenhaber. More value from your absolute value circuit. *Back Burner*, 44(04), April 2010.

[28] Ming-Jen Pan and Clive A. Randall. A brief introduction to ceramic capacitors. *Electrical Insulation Magazine, IEEE*, 26(3):44–50, May 2010.

[29] Winfield Hill Paul Horowitz. *The Art of Electronics*. Cambridge University Press, 3 edition, 2015.

[30] Matthew Pilotte. Operation of rf detector products at low frequency: An-691, 2005.

[31] Ian Poole. Silver mica capacitor.

[32] S. Pooranchandra, B. Sasikala, and Afzal Khan. *Introduction to Electrical , Electronics and Communication Engineering*. FireWall Media, 2005.

[33] Krik K. Reed. Characterization of tantalum polymer capacitors, 2005.

[34] R.P. Sallen and E.L. Key. A practical method of designing rc active filters. *Circuit Theory, IRE Transactions on*, 2(1):74–85, March 1955.

[35] C.K. Sanathanan and J. Koerner. Transfer function synthesis as a ratio of two complex polynomials. *Automatic Control, IEEE Transactions on*, 8(1):56–58, Jan 1963.

[36] Sim surfing. Online.

[37] Sprague 50 year timeline.

[38] High voltage power supplies.

[39] Series ps300 high voltage power supplies.

[40] Electronic components - dipped mica capacitors.

[41] Webench filter designer.

[42] GE Welsch. Doped titanium for forming capacitors with high energy density. In *The 22nd Advanced Aerospace Materials and Processes (AeroMat) Conference and Exposition*. ASM, May 2011.

[43] Ceramic capacitor, November 2014.

[44] Electrolytic capacitor.

[45] Silver mica capacitor, January 2014.