EMBRY-RIDDLE
Aeronautical University™
SCHOLARLY COMMONS

Dissertations and Theses

12-2014

Langmuir Probe Instrument Suite for Mesosphere Turbulence Experiment Mission

Adam Blake
Embry-Riddle Aeronautical University - Daytona Beach

Follow this and additional works at: https://commons.erau.edu/edt

Part of the Engineering Physics Commons, and the Plasma and Beam Physics Commons

Scholarly Commons Citation

LANGMUIR PROBE INSTRUMENT SUITE FOR MESOSPHERE
TURBULENCE EXPERIMENT MISSION

BY

ADAM BLAKE

A Thesis

Submitted to the Department of Physical Sciences

and the Committee on Graduate Studies

In partial fulfillment of the requirements

for the degree of

Master of Science in Engineering Physics

LANGMUIR PROBE INSTRUMENT SUITE FOR MESOSPHERE TURBULENCE EXPERIMENT MISSION

by

Adam Blake

This thesis was prepared under the direction of the candidate's thesis committee chair, Dr. Aroh Barjatya, Department of Physical Sciences, and has been approved by the members of the thesis committee. It was submitted to the Department of Physical Sciences and was accepted in partial fulfillment of the requirements of the

Degree of

Master of Science in Engineering Physics

THESIS COMMITTEE:

_____

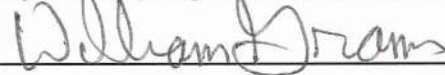Dr. Aroh Barjatya, Chair

_____

Dr. Charles Lee, Member

_____

Dr. Bereket Berhane, Member

_____

Dr. Peter Erdman, MSEP Graduate Program Coordinator

_____

Dr. Terry Oswalt, Department Chair, Physical Sciences

_____

William Grams, Ph.D. Dean, College of Arts & Sciences

_____ 12-3-14

Dr. Robert Oxley, Associate V.P. for Academics

iii

Abstract

The Langmuir probe technique is the predominant in-situ plasma diagnostic technique, and is arguably the only measurement technique that is universally flown on every sounding rocket investigation. Earth's mesosphere region (80-120 km) is a host of many dynamic phenomena such as the noctilucent clouds, breaking gravity waves, inversion layers, settlement of mesospheric smoke particles, etc. As such it is critical to have a comprehensive suite of plasma diagnostics that can unambiguously measure various plasma parameters.

This thesis deals with the development and testing of three different Langmuir probe implementations suitable for investigation of the mesosphere which are to be launched as part of the Mesosphere Turbulence Experiment sounding rockets. Two of the implementations are fixed bias Langmuir probes with different probe geometries, and the third implementation is a typical cylindrical sweeping Langmuir probe.

Amongst the fixed bias Langmuir probe implementations, the multi-needle fixed bias Langmuir probe is a collection of three fixed bias needles at distinct voltages in the electron saturation region that can together make an absolute electron density measurement that is immune to low levels of spacecraft charging. Similarly, the multi-surface fixed bias Langmuir probe is a collection of three spherical probes, however these are biased at the same potential in the electron saturation region while each having a different metal electroplated on its surface. This probe not only gives relative plasma density measurements, but is also able to detect the presence of mesospheric smoke particles. The sweeping Langmuir probe is a traditional Langmuir probe implementation capable of giving us absolute density and electron temperature along with payload floating potential.

The work towards this thesis involved the design of these various implementations in National Instrument's Multisim, the layout of the boards in National Instrument's Ultiboard, the board population, calibration and testing. Finally, the sweeping Langmuir probe electronics were also tested in the new ERAU Space Plasma Chamber.

Acknowledgments

Contents

List of Tables

List of Figures

Nomenclature

ADC Analog to Digital Converter

CONE Combined measurement of the Neutrals and Electrons

DAC Digital to Analog Converter

GPIO General Purpose Input/Output

MCX micro coaxial

mDCP+ multi-surface fixed bias DC Langmuir Probe

mDCP- multi-needle fixed bias DC Langmuir Probe

MTeX Mesosphere Turbulence Experiment

PCB Printed Circuit Board

SLP   Sweeping Langmuir Probe

SPI   Serial Peripheral Interface

SWaP size, weight and power

TQFP Thin Quad Flat Package

USART Universal Asynchronous Receiver and Transmitter

Chapter 1

INTRODUCTION

Earth's altitude range of 80 km to 1000 km is made up of ionized gases, a plasma, and is called the ionosphere. The region between 80 km - 120 km is a host to many dynamic phenomena as it is the interface between the neutral dominant altitudes below 80 km and plasma dominant altitudes above 120 km. Some of these phenomena include gravity wave breaking, sporadic-E layers, sudden metal layers, noctilucent clouds, mesospheric inversion layers, etc. In fact this altitude region is a unique dusty plasma laboratory as it constitutes mesospheric smoke particles that are formed when incoming meteors ablate and settle within this altitude region. Due to low altitudes, this region cannot be investigated in-situ by satellites. Thus, the only mechanism for in-situ investigation of this altitude region is through sounding rockets. One of the ubiquitously used measurement techniques for in-situ plasma diagnostics on sounding rockets is that of Langmuir probes.

Within this chapter, we first introduce the background of Langmuir probes followed by a background of the Mesosphere Turbulence Experiment (MTeX) sounding rockets. The subsequent chapters detail the design and calibration of the various Langmuir probes, followed by the testing results within a plasma chamber.

1.1   Langmuir Probe Review

The use of electric probes in a plasma chamber was pioneered by Irving Langmuir almost a century ago (*Mott-Smith and Langmuir*, 1924, 1926). He was the first to use electric probes to measure plasma density, temperature and potential in plasma chambers. Consequently, electric probes are also commonly known as Langmuir probes. With the development of the space age, Langmuir probes have been used on countless rockets, satellites and interplanetary spacecrafts to perform in-situ measurements of electron density (n_e) and temperature (T_e), ion density (n_i), and as an indicator for spacecraft charging (*Brace*, 1998; *Ferguson et al.*, 2001).

A Langmuir probe consists of measuring the current through a conducting metal collector that is biased at a known voltage. The resulting current-voltage (I-V) curve is typically of the form shown in Figure 1.1 (*Barjatya*, 2007). Note that the current "from" the probe to the plasma (i.e. electron collection current) is considered positive. Within the figure, ion current is exaggerated by an order to magnitude.
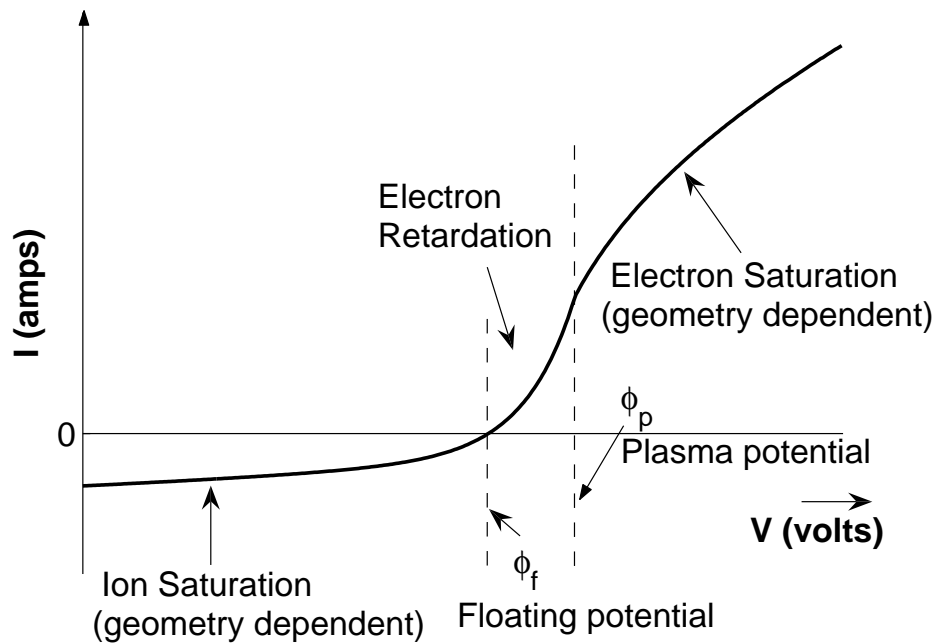
Figure 1.1: Expected I-V Curve from a Langmuir Probe

In a complicated plasma environment, the probe current could be the sum of various currents such as thermal currents, photoelectron current, secondary electron current, backscatter electron current, etc. However, in a simple environment: night-time ionosphere with no active aurora, the predominant current to a Langmuir probe is the thermal current. This current is a direct result of the maxwellian distribution of electrons and ions, and their resulting collisions (and collection) with the probe surface owing to their temperature. This current is given by Equation 1.1 (*Barjatya*, 2007).

$$I_{thermal,s} = \frac{1}{2} n_s q_s A v_x = n_s q_s A \sqrt{\frac{K_B T_s}{2 \pi m_s}} \tag{1.1}$$

where

A	Surface area	T_s	species temperature
n_s	species density	v_x	average velocity in certain direction
q_s	species charge	K_B	Boltzmann Constant
m_s	species mass		

The current of each species collected at the plasma potential is given exactly by the thermal current expression. At this potential there are no fields within the plasma and all species hitting the probe surface are collected. Due to their lighter mass, electrons have a higher thermal velocity and are thus the predominant collected species at the plasma potential. As the voltage is varied above or below the plasma potential, the collected current differs from the thermal current. At the floating potential, the total collected current is zero, i.e. collected electrons and ions are exactly equal. Between the floating potential and the plasma potential is the electron retardation region. In this region, the collected electron current is the predominant current as shown in Equation 1.2.

Outside of the electron retardation region the current enters saturation regions. When the applied probe bias is more negative than the floating potential then ion current is dominant and the region is called the ion saturation region. Above the plasma potential is the electron saturation region. Each of these two regions are

probe geometry dependent and the current collection expression is given by Equation 1.3.

$$I_e(\phi) = I_{thermal,e} \exp \frac{e(\phi - \phi_p)}{K_B T_e} \tag{1.2}$$

where

ϕ probe bias ϕ_p plasma potential
T_e electron temperature K_B Boltzmann Constant

$$I_s(\phi) = I_{thermal,s} \left(1 + \frac{q_s(\phi - \phi_p)}{K_B T_s} \right)^{\beta} \tag{1.3}$$

where

$$\beta = 0 \text{ for Flat Plate Probe}$$

$$\beta = 1/2 \text{ for Cylindrical Probe}$$

$$\beta = 1 \text{ for Spherical Probe}$$

The electron retardation region is used to derive the electron temperature. Taking the log of the current in the electron retardation region gives us

$$Ln(I_e(\phi)) = \frac{e\phi}{KT_e} + Ln(I_{thermal,i}) \tag{1.4}$$

By plotting the natural log of the probe current vs. the probe voltage, electron temperature can be calculated using

$$T_e = \frac{e}{(slope) * K} \tag{1.5}$$

The plasma density can consequently be calculated from the intercept at the plasma potential using

$$T_i = \frac{\exp(intercept)}{q_i A} \sqrt{\frac{2\pi m_i}{K_B T_i}} \tag{1.6}$$

The simplest Langmuir probe circuit is that using an operational amplifier (op-amp) in a transimpedance amplifier configuration, schematically shown in Figure 1.2.
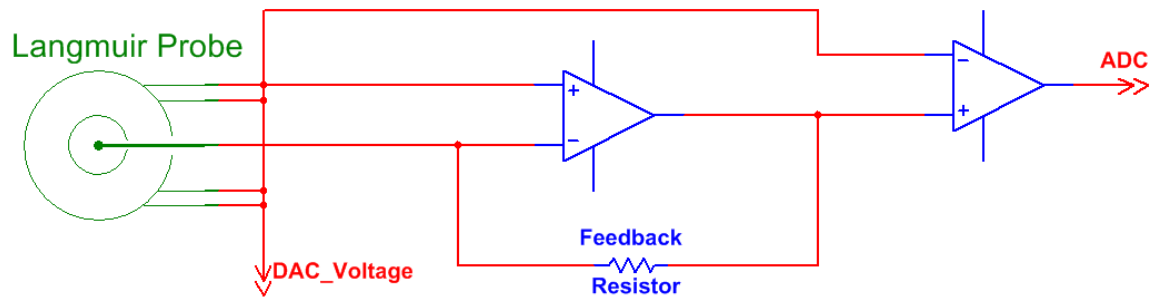
Figure 1.2: Basic Langmuir Probe Circuit

The transimpedance amplifier is a commonly used circuit to convert very low level currents to a measurable voltage. A stable and precise voltage is applied to the non-inverting terminal of an op-amp that has a very low bias current. Due to the "virtual ground" property of the op-amp, the inverting terminal operates at the exact same voltage. This inverting terminal is connected to the probe surface, where the applied potential results in current collection from the plasma. Due to the very high input impedance of a good op-amp, this collected current is forced to flow through the feedback resistor. Thus, the output of the transimpedance amplifier is a voltage sum of the applied bias and the voltage drop across the feedback resistor due to the collected plasma current. This output voltage of the transimpedance amplifier is then input to a difference amplifier to subtract the bias voltage, leaving simply the voltage drop across the feedback resistor, which is now an indicator of collected current.

The Langmuir probe is typically implemented in two ways. The most common way: sweeping Langmuir probe, is where a voltage is swept from some negative voltage to a positive voltage and a complete I-V curve is obtained. Thereby, using the method described above, one can derive plasma density, temperature and spacecraft floating potential a.k.a. spacecraft charging. However, completing a sweep takes time and thus the spatial resolution of the derived parameters is low. A second implementation of the Langmuir probe is where the probe is continuously biased at a fixed potential in the electron or ion saturation region. The current in either saturation region is directly proportional to the plasma density. Thus, the high cadence measurement of the saturation current gives a high spatial resolution measurement of relative plasma

density that then needs to be normalized to another absolute density measurement source. The MTeX Langmuir probe suite does both of these implementations.

1.2 Mesosphere Turbulence Experiment (MTeX) Mission Overview

The goal of MTeX is to answer the specific question: What is the contribution of wave generated turbulence to the energetics and mixing in the mesosphere lower-thermosphere in the presence of persistent regions of stability and instability? The observations will be made in a persistent environment that will frame the simulation studies that provide closure. The investigation also addresses the broader question of wave-wave interactions where larger scale waves routinely create regions of stability and instability for smaller-scale waves. MTeX's effort will also contribute towards answering one of the key questions in solar-terrestrial science: How do meteorological processes control the impact of solar processes on the Earth's atmosphere? This meteorological control is critical in understanding our atmosphere's response to auroral, radiation belt, and solar energetic particles, and associated effects on nitric oxide and ozone. This topic is critical to NSF's aeronomy community.

Recent observations have highlighted how the middle and upper atmosphere circulation modulates the impact of solar processes on the atmosphere. These observations raise questions about turbulent transport in the atmosphere and how we understand the dual processes of turbulent heating and turbulent transport. Measurements show wide variation in turbulent energy dissipation and eddy diffusion. Wintertime planetary wave activity provides persistent (over periods of hours to days) environmental conditions that support wave instability and generation of turbulence. Understanding these processes requires high resolution fluid dynamic simulations that capture the instability and turbulent structures.

MTeX's approach is to measure turbulence at meter scales in a well-defined persistent environment where the background stability can be characterized at km scales

and the temporal evolution of the environment and associated wave activity is documented at scales of 10's of minutes. The MTeX team will then use these measurements to frame fluid dynamic simulations where we can resolve the characteristics of breaking waves and turbulence and determine the characteristics of turbulence generation, dissipation, and diffusion in the middle and upper atmosphere.

The MTeX mission will yield high resolution measurements of neutral density, temperature, electron density profiles, and turbulent fluctuations to characterize turbulence and associated instabilities in the presence of a persistent mesosphere inversion layer in the 70-90 km altitude region. MTeX will launch two similarly instrumented rockets in January 2015 from Poker Flats Research Range in Alaska. A ground based Rayleigh LIDAR will be used to monitor gravity wave activity in the stratosphere, mesosphere and lower thermosphere at altitudes of 40-90 km. The event causing the mesospheric inversion layer should last a few hours; therefore a second rocket will be launched within one hour of the first launch. The data will then be used to model/simulate the dissipation rates and turbulence involved with these events.

The two instrumented rocket payloads will include both a neutral sensor suite, to characterize neutral turbulence, and a plasma sensor suite, to characterize plasma turbulence. The neutral sensor includes a Combined measurement of the Neutrals and Electrons (CONE) ionization gauge being provided to the mission by Institute of Atmospheric Physics, Germany. The plasma sensor suite includes three fixed-bias multi-surface Langmuir probes, one sweeping Langmuir probe, three fixed bias multi-needle Langmuir probes and an RF impedance probe. The design, calibration and testing of the various Langmuir probes are the topics of this thesis and are presented in the subsequent sections. A picture of one of the assembled payloads with booms deployed is shown in Figure 1.3.
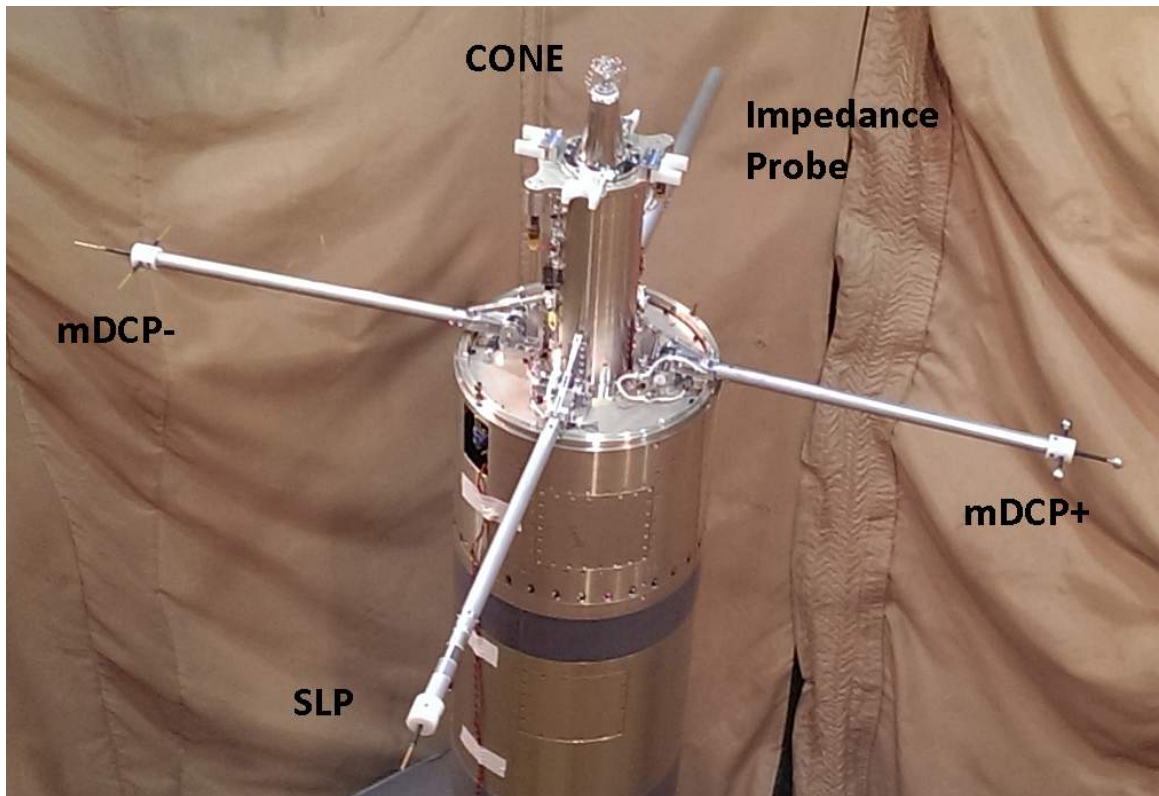
Figure 1.3: MTeX Payload Boom Deployment

Chapter 2

MTeX Langmuir Probe Instrument Suite

As described in the introduction section, Langmuir probes are typically implemented in two ways: sweeping Langmuir probes and fixed bias DC Langmuir probes. On a sounding rocket platform, the former gives low spatial resolution measurements of plasma density, electron temperature and spacecraft floating potential. The latter implementation gives high spatial resolution relative plasma density measurements. The MTeX Langmuir probe suite has various implementations of each.

For MTeX the first implementation is a traditional cylindrical sweeping Langmuir Probe (SLP). The other two implementations are a unique twist on the fixed bias Langmuir probe technique. The multi-surface fixed bias DC Langmuir probe (mDCP+) has three spheres each coated with a different metallic surface biased at the same potential in the electron saturation region. And the multi-Needle fixed bias DC Langmuir probe (mDCP-) has three identical needles each biased at different potentials within the electron saturation region. All three of these implementations use the same op-amps, Analog to Digital Converter (ADC), micro controller and power conditioning boards. In this chapter we first detail the common design features of these boards and then their unique features.

Each of the Langmuir probe implementations is broken down physically into two separate boards. The front end boards carry the analog circuitry as well as the digital

electronics for talking to the rocket telemetry system. The power board contains the power conditioning circuits that convert the unregulated battery voltage to various needed regulated voltages. Each front-end board is 6.3cm x 6.1cm and the power board is 6.1cm x 4.3cm. The front end and power board combination is contained in an aluminum box that is 12.8cm by 8.2cm with metallic separation between each of the boards effectively giving each board its own faraday cage. While the work towards this thesis involved the design, layout and population of the power boards as well, those designs will not be discussed. The power board design and layouts are simply presented in the appendix. Similarly, the mechanical design and fabrication of the booms and the probes themselves were done by Finn Carlsvi, Benjamin Wallace, and Michael Arsenault. Those mechanical designs are not included as part of this thesis. However, pictures of the final booms and probe heads are shown here for completeness.

2.1 General Block Diagram

Each of the Langmuir probe implementations has a basic structure for the front end as shown in Figure 2.1. The transimpedance amplifier applies a known bias to the probe and converts the collected current from the probe into voltage. The difference
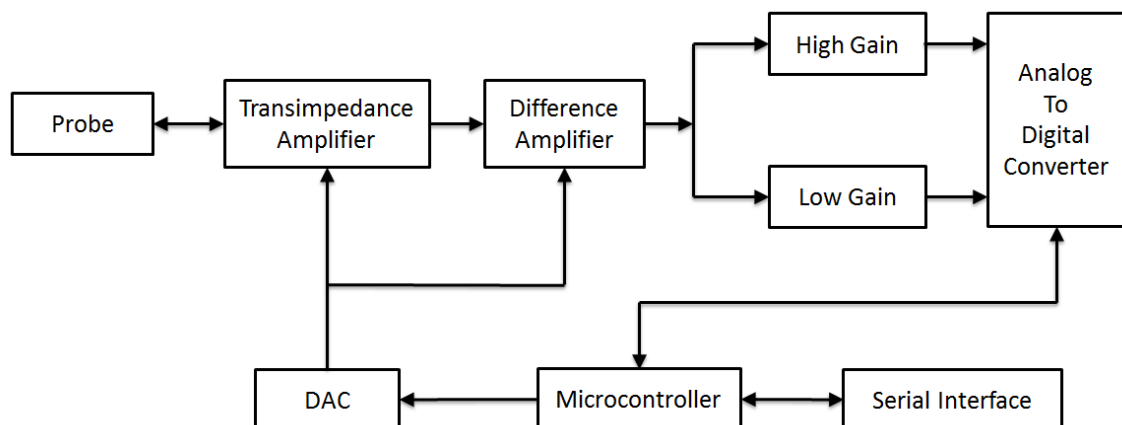

Figure 2.1: Payload Layout

amplifier subtracts the applied bias from the total output of the transimpedance amplifier leaving only the voltage that is a result of collected current. The output of the difference amplifier is fed to two separate amplifiers providing a low gain and a high gain channel. These two channels are then sampled through the analog to digital converter by the microcontroller and subsequently transmitted out.

2.2 Part Choices

Every chip in the design was carefully chosen for a specific reason. In this section we detail our part choices and the part's performance.

2.2.1 Front End Op-Amp

The ratio of spacecraft surface area to Langmuir probe surface area needs to be at least a few thousand times (*Szuszczewicz*, 1972). As the rocket payloads are fairly small, the probe dimensions have to be really tiny. The probe collection current is directly proportional to the probe area and the density. As the probe area is required to be tiny, in order for the measurements to succeed in low density plasma, the bias currents in the front end op-amp has to be in the pA - fA range. An op-amp with such low bias currents can convert low magnitude collected current to measurable voltage using a large feedback resistor in a transimpedance amplifier configuration.

During the initial stages of the project, LMC6041 was chosen as the front end amplifier due to its bipolar operation as well as its 2 fA bias current. While unipolar op-amps would have worked for the fixed bias Langmuir probes, they would not have worked for the sweeping Langmuir probe as the voltage applied ranges from ion saturation region to electron saturation region. Though the LMC6041 was designed for single supply operation, it can handle dual-supply operation.

The problem that was run into with the LMC6041 was that when run as bipolar, the output range was limited. In addition to this problem, the LMC6041 drew too much current from its negative power supply, making that power supply unstable. Therefore, midway through the project we selected a new op-amp: AD8627. It is

bipolar with rail-to-rail output but its bias current is 1pA. While this bias current is 3 orders of magnitudes greater than LMC604, it is still a factor of 10 smaller than the lowest current we would want to observe using our Langmuir probes. Additionally, the increased voltage output range will allow for larger sweeps in the sweeping Langmuir probe.

2.2.2 Difference and Gain Amplifier

As shown in the block diagram (Figure 2.1), following the transimpedance amplifier we need a difference amplifier to subtract the applied bias voltage and then two gain stages. In past Langmuir probe designs built at Embry-Riddle, this was accomplished using three separate op-amps: one in difference amplifier configuration, and then two in non-inverting amplifier configuration. This design has proven to be noisy due to more traces on the Printed Circuit Board (PCB) as well as having three separate chips also adds noise. Thus, for the MTex Langmuir probe suite, we chose to use an instrumentation amplifier. While an instrumentation amplifier is essentially made up of three op-amps, the benefit is that all components are within the same chip thereby potentially reducing the noise. Additionally our chosen chip, INA2128, has two inbuilt instrumentation amplifiers thereby providing us with two gain channels all in one single chip. The model of the INA2128 can be seen in Figure 2.2 (*TexasInstruments*, 2007).

2.2.3 Microcontroller

The microcontroller was chosen from the Atmel family due to previous experience with Atmel interface and programming as well as availability of excellent product selection with low SWaP (size, weight and power). The microcontroller chosen was the AT32UC3C2512C shown mounted on the SLP board in Figure 2.3. It is from Atmel's 32 bits architecture series and has 3 different operating frequencies with the max operating frequency being 66MHz. It comes in the 64pin Thin Quad Flat Package (TQFP) which is 12mm x 12mm (*AtmelCorporation*, 2012).

Our chosen package only has 45 general-purpose input/output (GPIO) pins versus
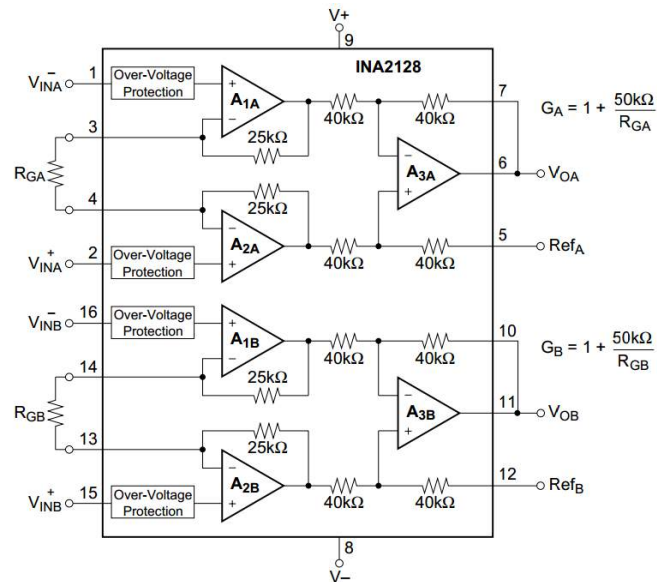
Figure 2.2: INA2128 Design

the LQFP144 and the TQFP100 packages which have 123 and 81 GPIO pins respectively. While less GPIO pins than the other options it still has 8 external interrupts, 1 serial peripheral interface (SPI), and 4 Universal asynchronous receiver and transmitter (USART) which can also be run as SPI masters. It has a 12-bit ADC with 11 channels that can be used for housekeeping and a 12-bit Digital to Analog Converter (DAC). These capabilities fit all our needs for speed and mandatory components that we need to communicate with. It has the maximum memory of 512 KB of Flash Ram.

Figure 2.3: AT32UC3C2512C Populated on SLP Board

2.2.4   Analog to Digital Converter

The ADC selected was the Analog Devices AD7656-1. It includes 6 individual 16-bit ADCs, each with an individual sampling rate of 250KSPS. All 6 channels are sampled simultaneously (*AnalogDevices*, 2012).

The AD7656-1 is the upgraded version of the AD7656 that requires less decoupling and has lower noise. The AD7656-1 comes in a 64-pin TQFP package, the identical package that our chosen microcontroller comes in, which simplified our breakout board requirements during testing. Its standard communication is to be run in parallel mode, but through hardware configurations it can be set for SPI communication. This setup is desired because everything is set before hand in the hardware. All of the channel select lines are tied together, as having additional channels selected does not affect sampling speed. Thus, the desired channels are parsed in the microcontroller. For the ADC, the data is output after the busy goes low (meaning conversions are done) and on the falling edge of the SPI clock in accordance with Figure 2.5.
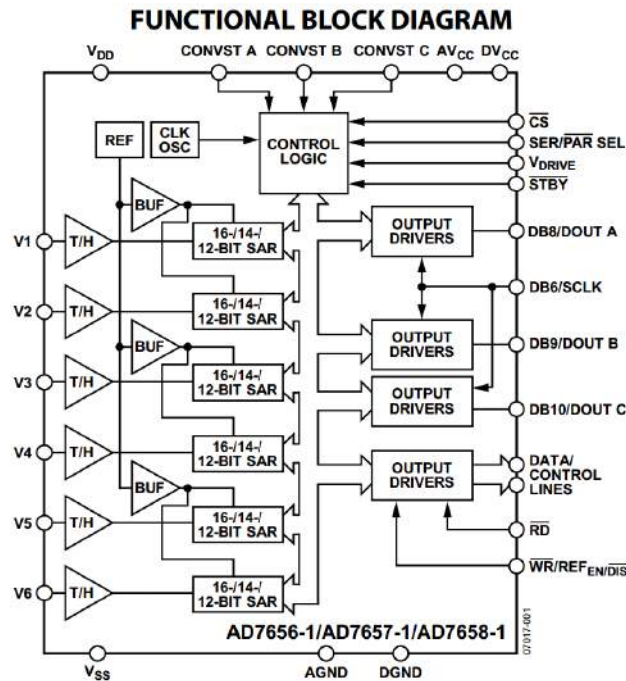

Figure 2.4: Front-end ADC
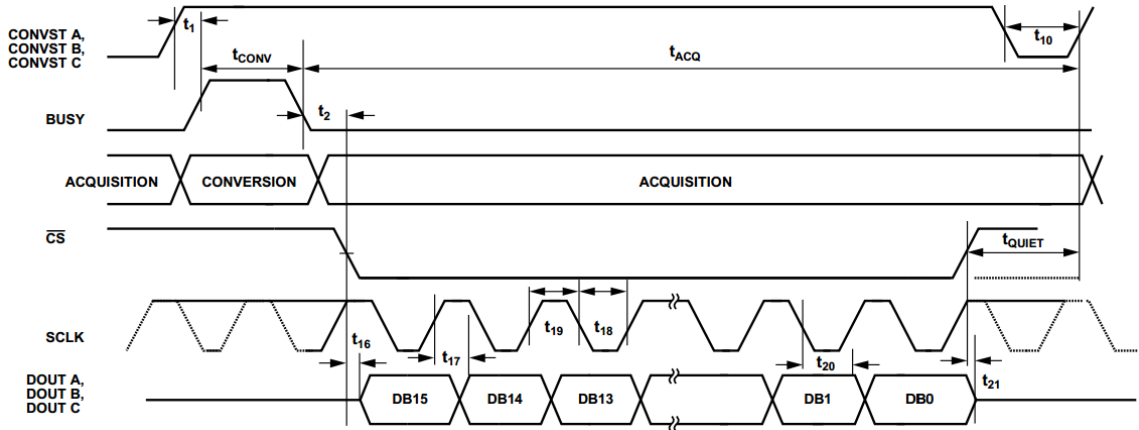
Figure 2.5: AD7656 Serial Interface

2.2.5   Digital to Analog Converter

For the flight mode of SLP, the probe bias voltage is swept from -1.5 V to +3.5 V. The easiest way is to use a DAC. The DAC needs to be able to handle both positive and negative voltages (bipolar) so that SLP can sweep through both the ion and electron saturation regions. The AT32UC3C2512C has an internal DAC that can be used to sweep. However, the problem is that it is only unipolar; sweeping from 0 to 5 volts. To fix this, basic amplifiers can be used to scale and shift this voltage. The INA129 instrumentation amplifier was chosen for this because of its low noise and bias current. Combined with a voltage reference the 12-bit DAC output then ranges from -3 volts to +4.5 volts.

However, when tested on a PCB, the noise in the internal DAC was just too high: over 1000 counts of noise using the 16-bit ADC discussed above. An external DAC was thus needed. The DAC eventually chosen was the DAC8581 (*TexasInstrument*, 2009). It is a 16-bit DAC that has +/- 5 volt range with fast slewing and settling time. This gives a stepsize of 0.153 millivolts. However for more accuracy, a lower volt reference chip could be used. If a 3.3 volt reference chip was used, this would achieve a stepsize of 0.092 millivolts.
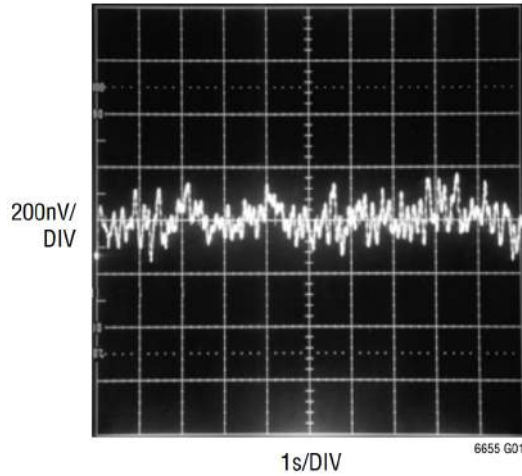
Figure 2.6: LTC6655 0.1Hz to 10Hz Noise Measures in 10 Second Sample Time
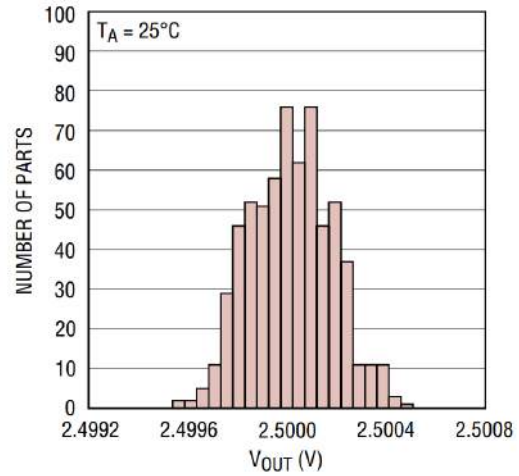
Figure 2.7: 2.5 Vout Distribution

2.2.6 Voltage References

All 3 boards, the SLP, the mDCP+ and the mDCP-, have voltage references on them for driving the probe potential. All come from the Linear Technology LTC6655 family. Each of the chips in the family has low drift, low noise and high accuracy. The family is especially suited for Langmuir probes due to wide values of voltage references available, all while maintaining same package and same pinout. The LTC6655 family is accurate to 0.025% with a maximum line regulation of 40 ppm/V, as shown in Figure 2.7 and a maximum load regulation of 15 ppm/mA. These reference chips

Table 2.1: Voltage References per Board

Board	Chip	Voltage
SLP	LTC6655BHMS8-4.096	4.096 Volts
mDCP+	LTC6655BHMS8-3.3	3.3 Volts
mDCP-	LTC6655BHMS8-3.3 LTC6655BHMS8-4.096 LTC6655BHMS8-5.0	3.3 Volts 4.096 Volts 5.0 Volts

have an output voltage peak-to-peak noise of 0.25 ppm (775nV). The noise for the LTC6655 family is average over 1,000 10-second trials. One of these 10-second trials is shown in Figure 2.6 (*LinearTechnology*, 2009, 2014).

2.3   Sweeping Langmuir Probe

This board had a straightforward implementation of the block diagram shown in Figure 2.1. The detailed schematic and layout are shown in Appendix 5.1 and 5.2. A linear sweep will be used on the SLP for the MTEX launch. Because of the limited telemetry bandwidth, the SLP has been allocated 1,440 16-bit words in the telemetry matrix. This has been broken down into the 8 words needed for housekeeping and then 716 words for each of the high-gain and low-gain channels. The sweep will have step sizes of 56 counts per step, and run from a count of -12096 to + 28000. This is a sweep range of roughly -1.5 to +3.5 volts. The SLP uses a LTC6655BHMS8-4.096 chip to scale and offset the 16 bit DAC output which leads to a step size of 7 millivolts. The DAC output is calibrated for each of these exact steps. Similarly, the ADC output is calibrated over various temperature ranges and resistive loads. The measured I-V curve in-situ will then be analyzed post flight to get various plasma parameters such as electron density, electron temperature and payload floating potential.

The probe for the SLP is a gold plated needle with dimensions of 40 mm length and 1.5 mm diameter. It is shown in Figure 2.8. The guard is 5 mm in diameter and 5 mm in length, and is run at the same potential as the SLP.

2.4   Multi-Surface Fixed Bias DC Langmuir probe

This is the most important instrument for the MTeX mission. As the name implies, the multi-Surface fixed bias DC Langmuir probe (mDCP+) has three fixed bias Langmuir probes, each with a different surface material but same potential bias, located on the same boom to measure high resolution plasma collection current, which can then be interpreted as relative plasma density measurement. Each sphere has a diameter of 0.75 inches. MaCWAVE-project conducted in the years 2002 from Andoya

Figure 2.8: Boom for the SLP and Ground-LP

Rocket Range and 2003 from the Swedish ESRANGE (*Goldberg et al.*, 2004, 2006) utilized rocket borne in-situ measurements of relative electron density fluctuations as a tracer for neutral air turbulence (*Croskey et al.*, 2004, 2006). It is now well established from both theory (*Kelley et al.*, 1987; *Cho*, 1992; *Rapp et al.*, 2003a) and observations (*Lubken et al.*, 1998; *Rapp et al.*, 2003b; *Strelnikova and Rapp*, 2009) that the presence of charged heavy particles significantly changes the power spectrum of plasma density fluctuations. Hence, the question arises whether electron density fluctuations (or plasma density fluctuations in general) can indeed generally be used as a tracer for neutral air turbulence outside regions with charged heavy ice particles as assumed in *Croskey et al.* (2006) and many other previous studies (*Blix et al.*, 1990). The multi-surface Langmuir probe being flown on MTeX has a unique capability of detecting the presence of meteoric particle layers along the rocket trajectory and thus aiding in elimination of suspect turbulence data from plasma density fluctuation measurements.

The charging of metallic surfaces by charge transfer from dust particles, due to differences in work functions or due to frictional contact, is known as triboelectric charging. If two surfaces come merely in contact with each other and then separate, the surface with lower work function loses an electron to the surface with higher work function (*Harper*, 1967). This triboelectric current to a surface moving in dusty plasma is in addition to any thermal plasma current. The triboelectric current from neutral dust or neutral meteoric smoke particles to a fixed-bias Langmuir probe can

also be used to determine a crude estimate of the particle number density and their primary metallic content.

The three individual sensors of the mDCP+ will be mounted on a boom at the payload fore and deployed radially outward from the payload long axis to reduce wake effects. A picture of the boom head is shown in Figure 2.9. The three spheres have had their surfaces coated with materials that have different work functions Platinum (5.9 eV), Rhodium (5 eV) and Indium (4.1 eV). Each of the spheres DCPs is biased +3.3 volts relative to the payload skin using a LTC6655BHMS8-3.3 chip. The schematic and layout are shown in Appendix 5.1 and 5.2. Essentially, the circuit of mDCP+ triplicates the SLP circuit on one single front end board, while using a single voltage reference chip. As a result, there are 6 channels of voltage: low gain and high gain from each probe. All six channels need to be sampled simultaneously so that the measurements are all made at the same location in plasma. Our chosen ADC is capable exactly of that.

Assuming the metallic composition of dust to be similar to that in meteorites (*Plane*, 2004; *McNeil et al.*, 2002), the dust particles will be composed of Potassium (2.29 eV), Sodium (2.36 eV), Calcium (2.87 eV), Magnesium (3.66 eV), and Iron (4.67 eV). All oxidized metals behave, as far as contact charging is concerned, like a
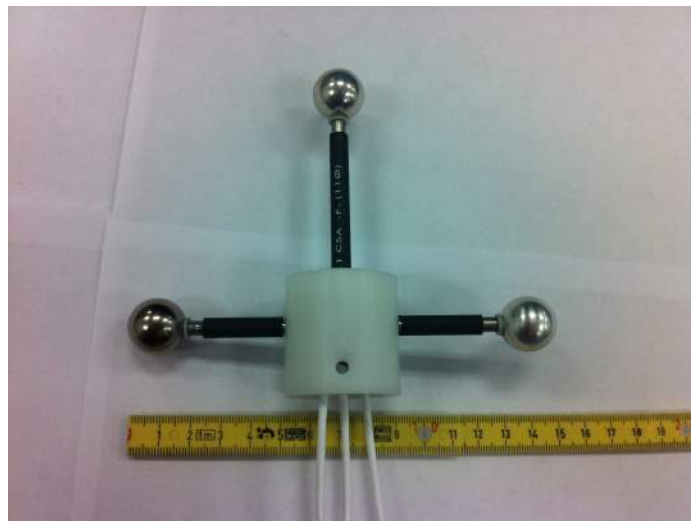

Figure 2.9: mDCP+ Boom

different metal with a work function equal to the depth of the acceptor levels in the adsorbed oxygen, which is about 5.5 eV and is largely independent of the nature of the metal (*Cabrera and Mott*, 1949; *Sternovsky et al.*, 2001). Although most of the constituents in the neutral smoke particles will be oxidized, the observations of sudden metal layers in the mesosphere imply the possible adsorption of unoxidized metals on the smoke/dust particle surface. These metals can be detected using the difference in triboelectric current among the 3 DCPs. The various metallic components of dust can be grouped into three categories depending on where their work function falls relative to that of the sphere surfaces. The presence of unoxidized iron, as a primary metallic constituent adsorbed on the smoke particles, will lead to deposition of electrons on the platinum and rhodium coated DCP and the acceptance of electrons from the indium coated DCP. This difference in triboelectric current will allow us to estimate the neutral dust particle density that is carrying unoxidized iron atoms. The presence of unoxidized metals with work function lower than 4.1 eV can be detected by little or no difference in triboelectric current to all three DCPs. The triboelectric current to the platinum coated DCP will give us total neutral smoke particle density, as even the oxidized particles will leave an electron on the high work function platinum DCP. The detection of meteoric charged particles along the rocket trajectory will allow us to identify spurious fluctuations in the ionization profile from true turbulent fluctuations.

2.5   Multi-Needle Fixed Bias DC Langmuir probe

As the name implies, the multi-Needle fixed bias probe uses multiple needles (cylindrical probes). Each needle is biased at a fixed voltage that is in the electron saturation region to get relative electron density measurement. The probes for the mDCP- are gold plated needles as well, and shown in Figure 2.10. Each needle is 50mm long and has 5 mm long guard running at the same potential as the probe needle.

Each of these three needles is at different potentials. As all needles are same in dimensions and mounted to the same boom, they essentially act as three simultaneous samples on the I-V curve in the electron saturation region at different applied biases.
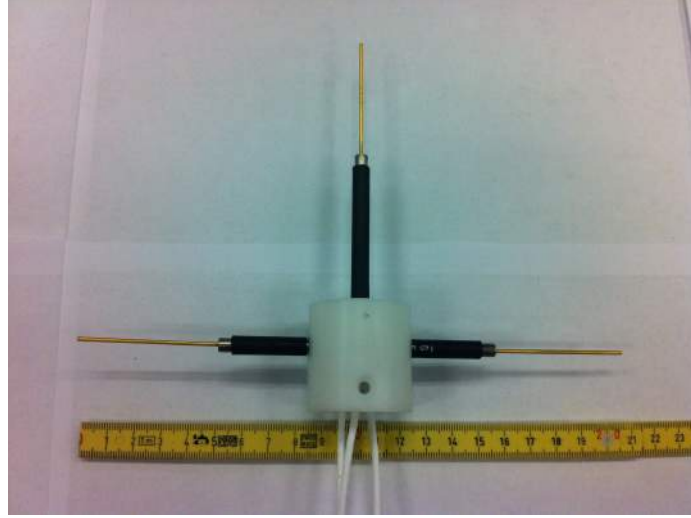
Figure 2.10: mDCP- Boom

The current collection expression in electron saturation region is given by Equation 1.3 with $\beta = 1/2$. It is now important to note that there is a linear relationship between square of the collected current and the applied bias. Thus, the difference in the square of currents measured at two different biases within the electron saturation region is give by

$$I_2^2 - I_1^2 = (Cn_eA_p)^2(V_{b2} - V_{b1}) \tag{2.1}$$

where C is a constant given by $\frac{e^{3/2}}{\pi}\sqrt{\frac{2}{m_e}}$. From this we get the following:

$$n_e = \frac{1}{CA_p}\sqrt{\frac{I_2^2 - I_1^2}{V_{b2} - V_{b1}}} \tag{2.2}$$

Eq. 2.2 gives us absolute density measurement from two fixed bias probe measurement (*Bekkeng*, 2009). Also note, that to get the absolute density only the relative separation of the two probe biases needs to be known, not the bias w.r.t. plasma potential. Therefore, this technique is essentially immune to spacecraft charging as long as the charging is not so severe that the operation of the needles is brought into the electron retardation region.

2.6   Design for Guarding and Powerplane Layouts

One of the major things needed for a low noise design is power planes. There should be dedicated power and ground planes. On all the front end boards, the third layer was used as a dedicated ground plane.

The basics of ground plane design are as follows:

1. Separate your analog and digital grounds. If possible create an analog ground plane and a digital ground plane that are connected with an inductor. As most of the front end board is analog, creating a slim digital ground plane that encompassed the microcontroller and communication chips without the ADC would have been impractical. One solution would have been to switch the microcontroller and ADC positions so that the digital section is better grouped. However, this would drastically increase the trace length of the ADC inputs adding noise to the signal.

2. Avoid running traces through the ground plane. This increases the impedance on the ground plane and interrupts the current flow. Current on a ground plane usually follows a return path that matches its sister trace above. Therefore, running traces can break up this flow.

3. If traces are run through the ground plane, minimize their length and above all make sure that you never cut the ground plane. When running these traces through the ground plane, route them around the outside of the board.

4. Watch where through-hole components and vias are placed. Through-hole connectors with a small enough pitch can also disrupt the ground plane. As shown in Figure 2.11, the through hole micro coaxial (MCX) connector, DE-15 and off-board connectors, are near the perimeter of the board.

5. Use voltage planes to protect sensitive signals. Each front end board uses a reference voltage that biases the attached probe. A power plane is added to this front end on every layer besides the ground plane as shown in Figure 2.12. The purpose of this is to protect a sensitive current signal (200 nAmps). Its
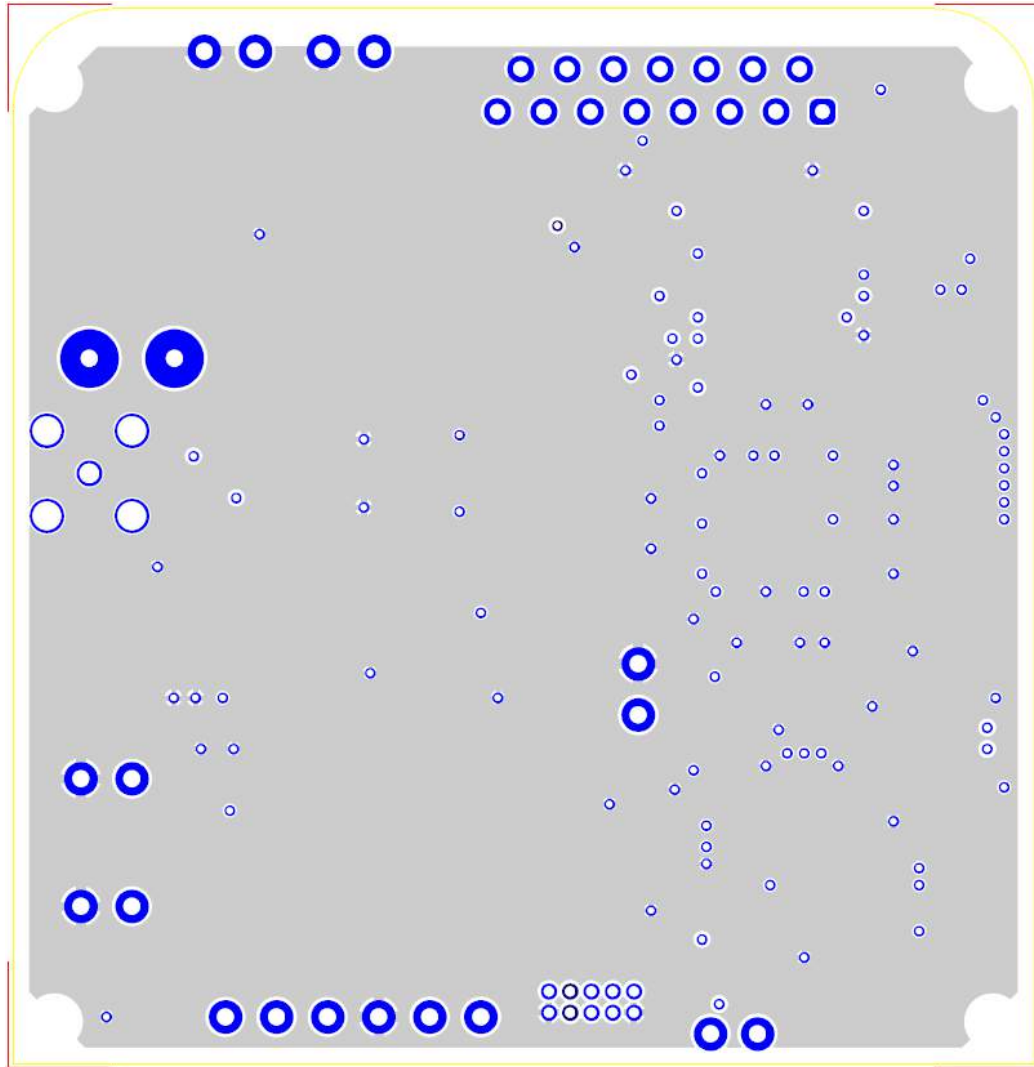
Figure 2.11: SLP 3rd Layer Ground Plane

trace goes from the middle of the MCX connector to pin 2 for the AD8627 to the feedback resistor. This power plane then extends to the input of the INA2128. The point of this is that everything around the signal is at this reference voltage, therefore reducing leakage into or out of the signal path.
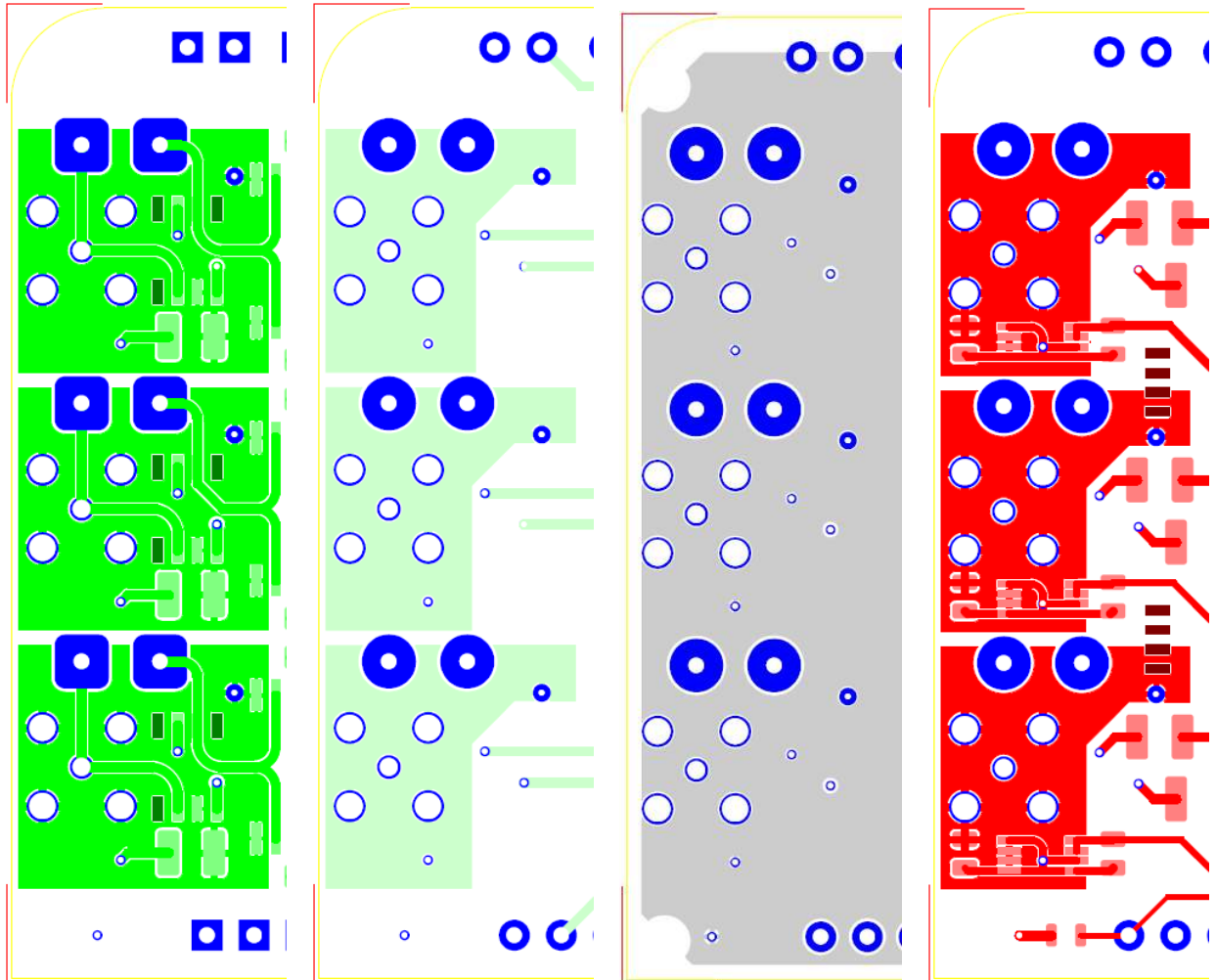
Figure 2.12: Guarding of the Front End
Top Layer -> Inner Layer 1 -> Inner Layer 2 -> Bottom Layer

2.7 Other Noise Reduction Techniques

The payload is divided into two sections to separate out the noise from the power board. By doing this, each board is in its own faraday cage. This is shown in Figure 2.13. To connect the boards, instead of running a single voltage wire, two wires are used to connect them. Each voltage is combined with a ground wire to form a twisted wire pair. The point of this is that a current on a wire forms a field. By using twisted

Figure 2.13: Payload Layout

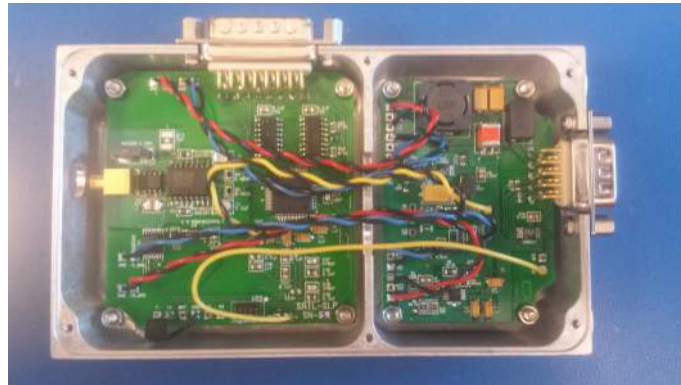wire pairs, this creates a canceling force that reduces noise added by connecting the individual wire.

Decoupling capacitors are also used on each incoming power line. The purpose of this is to prevent transmission of noise from one component of the circuit into another. The point is to isolate two circuits on a common line.

Chapter 3

Calibration and Testing

One of the most challenging things about the Langmuir probe suite fabrication was the soldering of the tiny parts on the small front end and power board PCBs. Over all, there are 81 parts on the front end board and 71 parts on the power board. After lots of trials, a procedure was developed wherein each major chip in the design was tested once populated. When the whole board was populated, each of the critical analog parts of the design were calibrated over temperature, i.e. the DAC/voltage references and the ADC. The details are presented in this chapter.

The testing will be done first on the major components and their calibration. Then the system as a whole will be tested in the Embry-Riddle Plasma Chamber.

3.1   External DAC Calibration

The external DAC on the SLP board is the DAC8581. This voltage output needs to be calibrated, just like the reference voltages would be on the fixed voltage boards. The best way to do this is to step the DAC and measure the voltage of the reference powerplane. This way, we know the exact voltage the probe is biased at at each step of the sweep.

3.1.1 Calibration Setup

The calibration setup is shown in Figure 3.1. A Matlab script steps the SLP electronics through the sweep steps. The applied voltage is then measured by the Tektronix digital multimeter DMM4050 and sent over a serial line to the computer which is recorded by the Matlab script before it steps to the next voltage. This measurement of the sweep steps is done over a range of temperatures to calibrate the DAC. The temperature of the electronics box is measured by the onboard temperature sensors that send the data over the serial line back to the computer. Thus, the Matlab script records the step number, the measured voltage by the multi-meter, and the temperature of the front end PCB, for each of the sweep steps. Data is recorded for an upsweep as well as a downsweep. As the payload will be launched from Alaska, we calibrate from room temperature down to freezing temperatures. The electronics are cooled using a dry ice pack. This whole calibration process can be automated. These codes can be seen in Appendix 5.3 & 5.4.

3.1.2 Calibration Results

The calibration data saved to a text file can then be analyzed. A sample snapshot of the data is shown in Table 3.1. The data includes the DAC voltage and its corresponding count. Each step also has housekeeping which includes the temperatures in counts and voltage monitors.
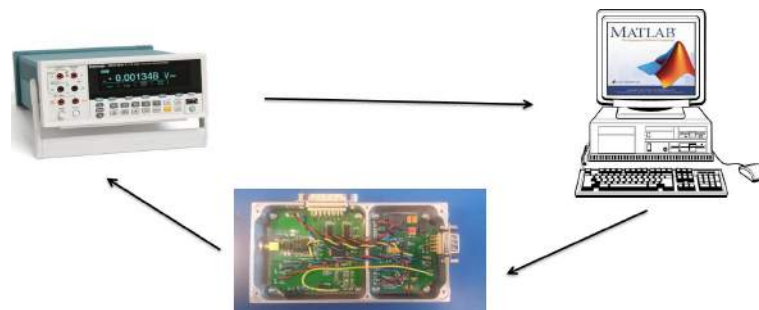

Figure 3.1: Setup for Calibration of the External DAC

Table 3.1: DAC Calibration Data

DMM4050 Voltage	DAC Count	12V ADC	5V ADC	Battery ADC	External Temp	Internal Temp 1	Internal Temp 2
-1.51192E+00	-14700	1045	354	2040	1763	1783	1778
-1.50520E+00	-14635	1045	352	2035	1762	1782	1779
-1.49861E+00	-14570	1048	362	2032	1766	1781	1779
-1.49197E+00	-14505	1039	362	2037	1763	1782	1780
-1.48529E+00	-14440	1043	355	2038	1763	1781	1780
-1.47867E+00	-14375	1046	353	2039	1763	1781	1778
-1.47192E+00	-14310	1047	359	2034	1764	1785	1778
-1.46506E+00	-14245	1047	350	2036	1764	1782	1780
-1.45833E+00	-14180	1044	359	2034	1765	1781	1778
-1.45181E+00	-14115	1048	363	2040	1765	1781	1777
-1.44523E+00	-14050	1046	356	2035	1764	1782	1780
-1.43844E+00	-13985	1046	351	2037	1762	1784	1780
-1.43160E+00	-13920	1040	357	2040	1762	1781	1778

This DAC can be calibrated in three ways. The first it to calibrate each sweep based solely on the reference voltage. This is shown in Figure 3.2. Using a calibration of this leads to a standard deviation of 1.118 mV. If temperature is calibrated in for each sweep, this yields a reference voltage error as shown in Figure 3.3. This only decreases the standard deviation to 1.112mV. While both these errors are within acceptable ranges, the third calibration method is more effective. This is to calibrate with respect to temperature but for each step point, not for the overall sweep. This is shown in Figure 3.4. This yields a standard deviation of 9.8 μV. These fits were done for the calibration data of MTeX payload 9420.

A polyfit can be applied to all of the sweeps done for different temperatures and yields equation 3.1

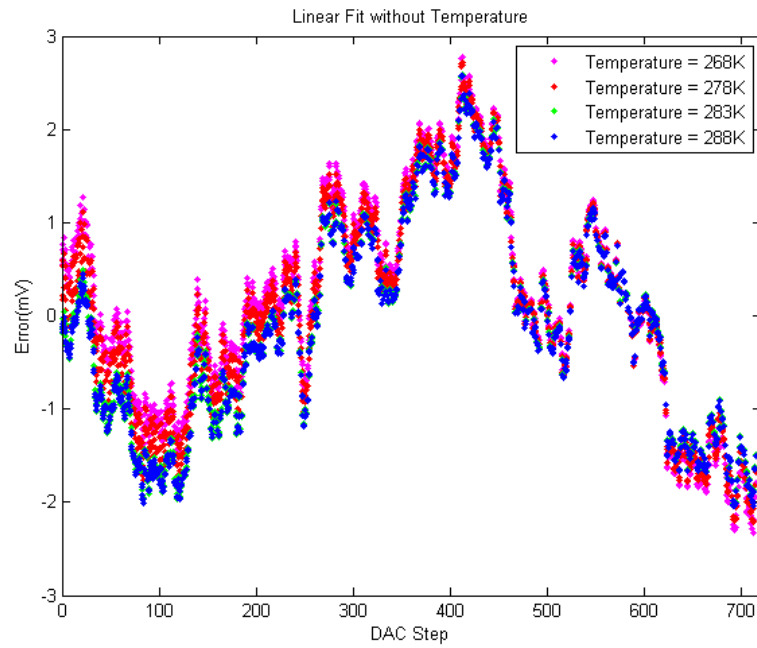$$V_{Reference} = -0.0072 * step + 3.5735 \tag{3.1}$$
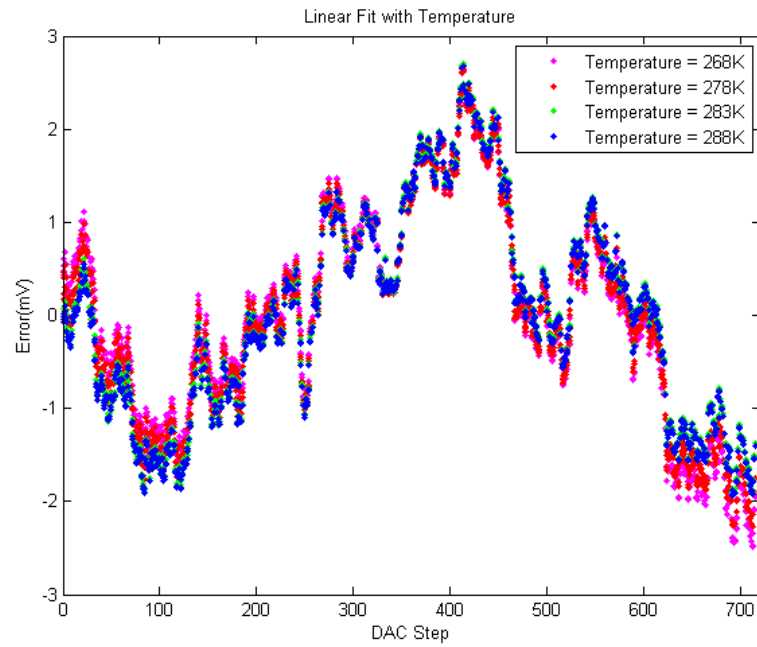
Figure 3.2: DAC Sweep Calibration

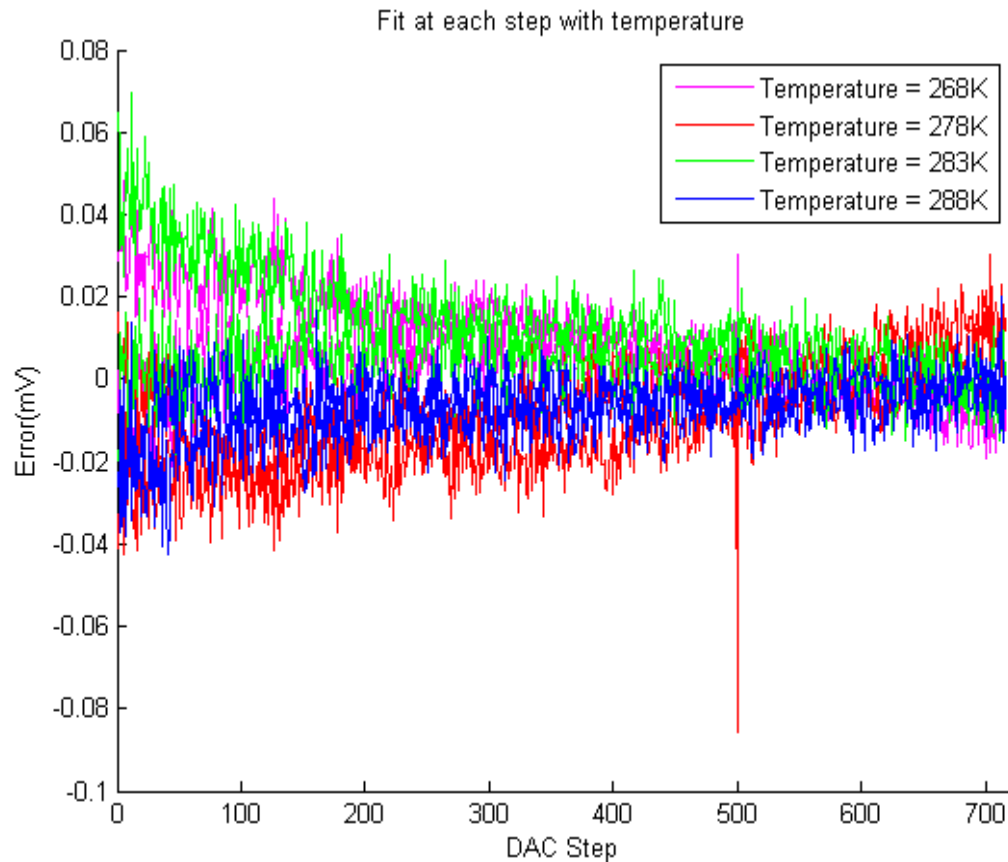Figure 3.3: DAC Calibration With Temperature

Figure 3.4: DAC Calibration With Temperature at Each Step

The decision comes down to the payload. For testing with the plasma chamber, the SLP will be at room temperature. Even with its internal heating, it reaches an equilibrium and a fit for the entire sweep is appropriate to use. However, for the MTeX launch, the payload can experience a wide range in temperatures due to the launch in Alaska. Therefore, a fit depending on temperatures and step is more appropriate for that.

3.2 External ADC

The external ADC on all the boards is the AD7656-1. It has 6 channels that are all sampled simultaneously, and each must be calibrated individually. The intent is to

correlate ADC counts to voltage and then relate that to the collected current by the probe.

3.2.1   Calibration Setup

The ADC calibration uses mostly the same code as the flight code with the only difference being that the flight code uses synchronous communication whereas the calibration code uses asynchronous communication. In order to find a relationship between the ADC counts and the collected current, we connect high precision resistors (+/- 0.5 %) to the probe connector. The resistors are thermally epoxied inside a large aluminum block as shown in Figure 3.5. The large metal block helps maintain thermal equilibrium, therefore keeping the resistance value very stable as the calibration is carried out. As we also know the resistor value, we can then calculate the exact current going into the front end amplifier. Thus, the measured current value can then be correlated to the ADC counts. The entire electronics box is placed inside a metal box that is grounded, in addition the electronics box being grounded as well, as shown in Figure 3.6l. Thus, we have a double faraday cage around the sensitive electronics. All grounds are connected. This metal box is then placed in a cooler with dry ice. As the SLP cools down, sweeps of data are taken. The sweeps are
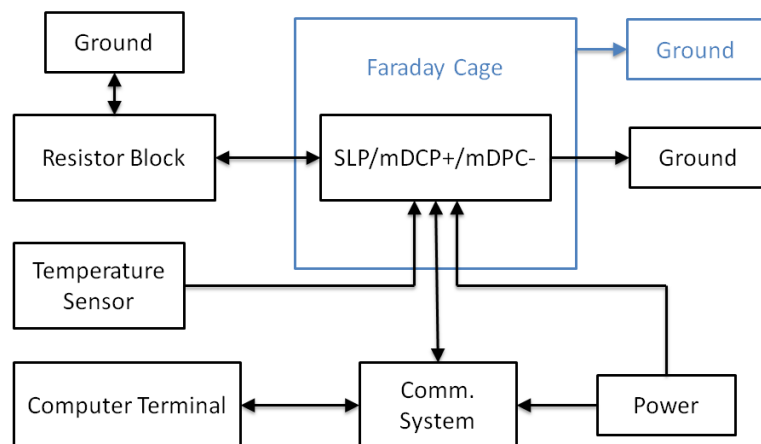

Figure 3.5: Load Resistors

Figure 3.6: ADC Calibration Set Up

Table 3.2: ADC Calibration Data

DAC Reference Step	Internal Temp 1	Internal Temp 2	External Temp	Low-Gain ADC Count	High-Gain ADC Count
422	1572	1579	1543	2837	28881
423	1574	1578	1543	2802	28518
424	1575	1577	1541	2766	28139
425	1575	1580	1541	2729	27742
426	1574	1579	1542	2689	27386
427	1575	1579	1544	2654	27011
428	1574	1579	1543	2615	26621
429	1575	1579	1543	2576	26257
430	1573	1577	1543	2544	25885

made using different resistor values such that the low gain channel and the high gain channel ADCs are fully exercised from a count of hundreds to above 65,000. At the very extremes of the 16 bit count range, the ADC is expected to be non-linear.

3.2.2 Calibration Results

The calibration data is saved to a text file that can then be analyzed. A sample is shown in Table 3.2. The data includes the DAC reference number and its corresponding ADC counts. Each step also has housekeeping which includes the temperature sensors. Each set of datafile contains a 716 step up sweep and a 716 step down sweep.

Most of the time a simple linear relation would be used for current and ADC counts. As shown in Figure 3.7, because the system is temperature dependent, it must be accounted for in the current voltage relationship. This yields Equation 3.2. These variables can be solved with the calibration data. The results for payload 9420 are shown in Table 3.3.

$$
\begin{aligned}
Current = K_{ADC} * Count_{VoltageADC} + K_{T1} * Count_{TempInt1} + \\
K_{T2} * Count_{TempInt2} + K_{Tex} * Count_{TempExt} + \\
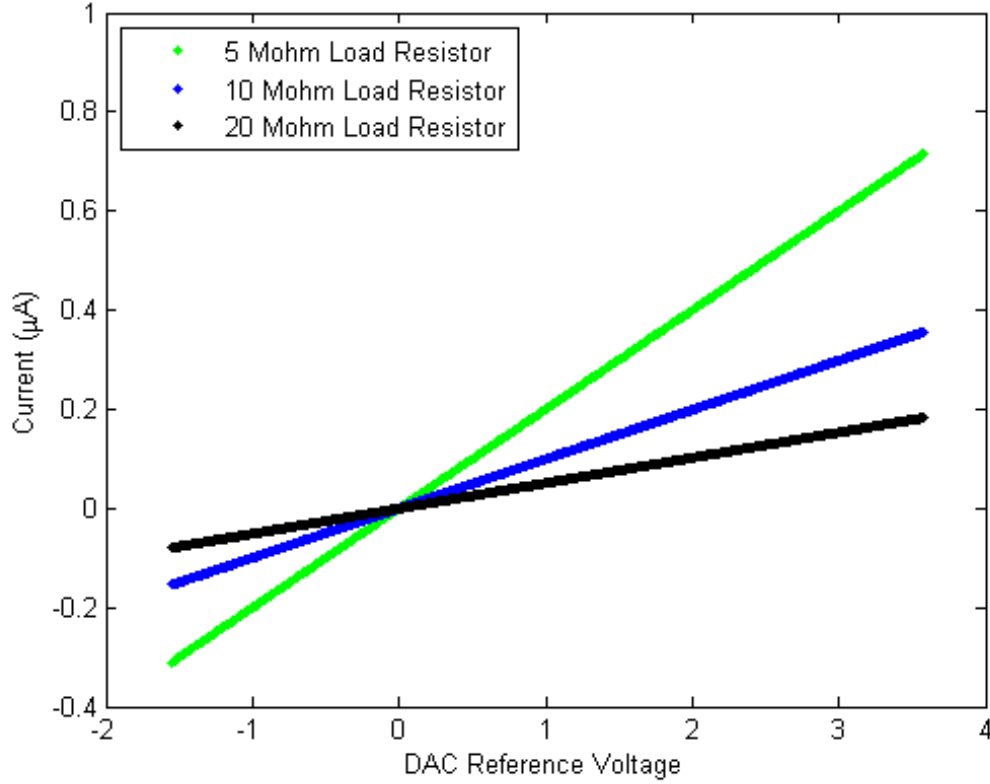K_{DAC} * Count_{DACindex} + K
\end{aligned}
\tag{3.2}
$$

Figure 3.7: ADC Calibration Curves

Table 3.3: ADC Calibration Constants for SLP Payload 9420

	Channel 1 Low Gain	Channel 2 High Gain
K_{ADC}	$+3.83951210^{-2} \frac{nA}{count}$	$+3.828632^{-03} \frac{nA}{count}$
K_{T1}	$-6.218336^{-03} \frac{nA}{count}$	$-8.724506^{-05} \frac{nA}{count}$
K_{T2}	$+1.672281^{-03} \frac{nA}{count}$	$-4.621282^{-05} \frac{nA}{count}$
K_{Tex}	$+3.198351^{-03} \frac{nA}{count}$	$-1.183654^{-04} \frac{nA}{count}$
K_{DAC}	$-1.559814^{-03} \frac{nA}{count}$	$-4.594986^{-05} \frac{nA}{count}$
K	$+4.541798$ nA	$+6.597717^{-01}$ nA
Standard Deviation	0.294452 nA	0.108505 nA

3.3   SLP testing in the Space Plasma Chamber

The ERAU Space Plasma Chamber was commissioned just in time for testing the
MTeX SLP electronics, as well as the MTeX needle probes. This section gives a brief
overview of the chamber and then presents some data taken with the SLP within the
plasma chamber.

3.3.1   Plasma Chamber Design and Operation

The plasma chamber is cylindrical with 2 m length and 1 m diameter. It was designed
and built by Z Lucky Enterprises to the specification provided by Embry-Riddle. It
has two pumps. The first is a roughing pump that can bring the chamber down to 0.5
Torr. The second is a turbo pump that runs at 45000 RPM and pumps the chamber
down to a base pressure of 10^{-7} Torr. Once the base pressure is reached, Argon is
introduced into the chamber to create a uniform ionized species.

   The plasma is created using hot filaments to ionize the Argon in the chamber. The
filaments are thoriated tungsten filaments. An electrical charge of approximately 40
V and a current around 25 A are applied to these filaments, as shown in Figure 3.10.
The filaments are then biased to a negative voltage to repel the created electrons
(*Lucky et al.*, 2014) and then pushed into the chamber. These filaments are shown in
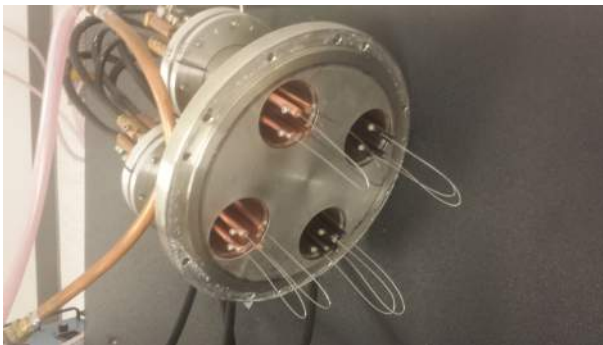Figure 3.8 and 3.9.

Figure 3.8: Thoriated Tungsten
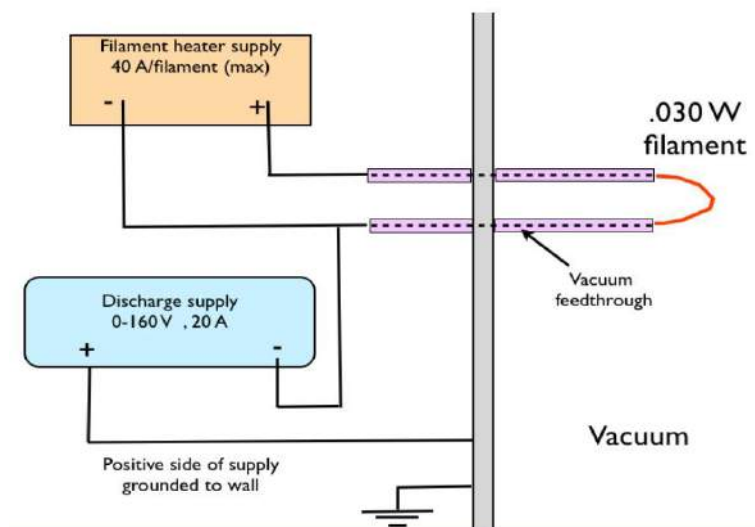Filaments

Figure 3.9: Active Filaments

Figure 3.10: Plasma Chamber Filament Configuration

3.3.2   Sweeping Langmuir probe testing results

The probe placed within the chamber is the exact dimensions as the one used for MTeX, but not electroplated. It is quite well known that Langmuir probes with contaminated surfaces give distorted I-V curves (*Oyama*, 1975). The distortion typically shows up in the sweep data as a hysteresis between upsweep and downsweep. The contamination creates a layer that can be modeled electrically as a parallel resistor and capacitor circuit.

When we put our needle probe for the first time in the chamber, we expected to see hysteresis. This is shown in Figure 3.11. Note that during this test, the density in the chamber was too high leading to instrument saturation in the electron saturation region. This includes a sweep up from -5.05V to 5.0V and then back down at 0.85 Hz. *Oyama* (1975) has pointed out that increasing the sweep frequency to greater than 10 Hz essentially bypasses the contamination capacitance and removes the hysteresis. Figure 3.12 shows the results of running the probe at higher sweep frequencies which bypasses the hysteresis. The figure shows six different frequency sweeps ranging from 2.02Hz up to 25.5Hz. In flight, we are running the SLP at 9.24 Hz. As can be seen, the hysteresis is very prominent for frequencies around 8Hz and below.
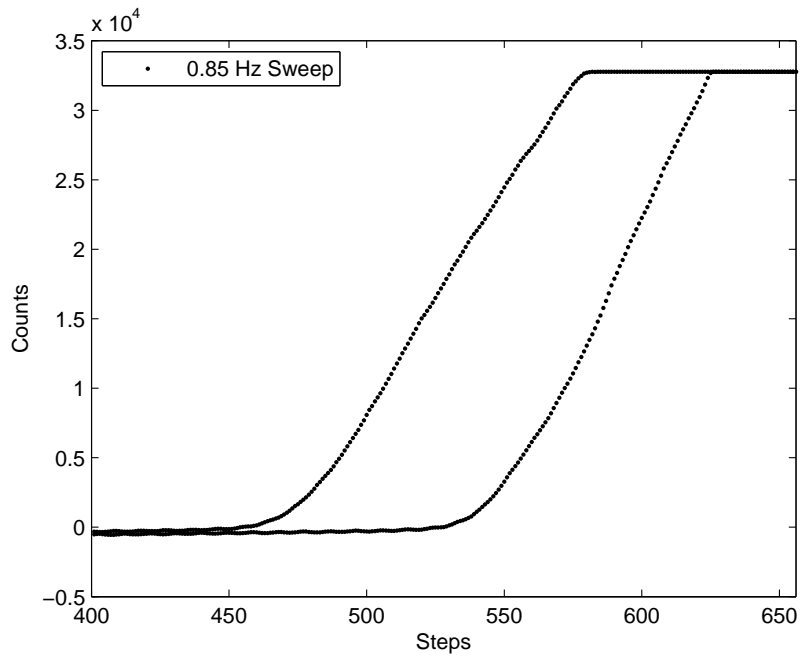
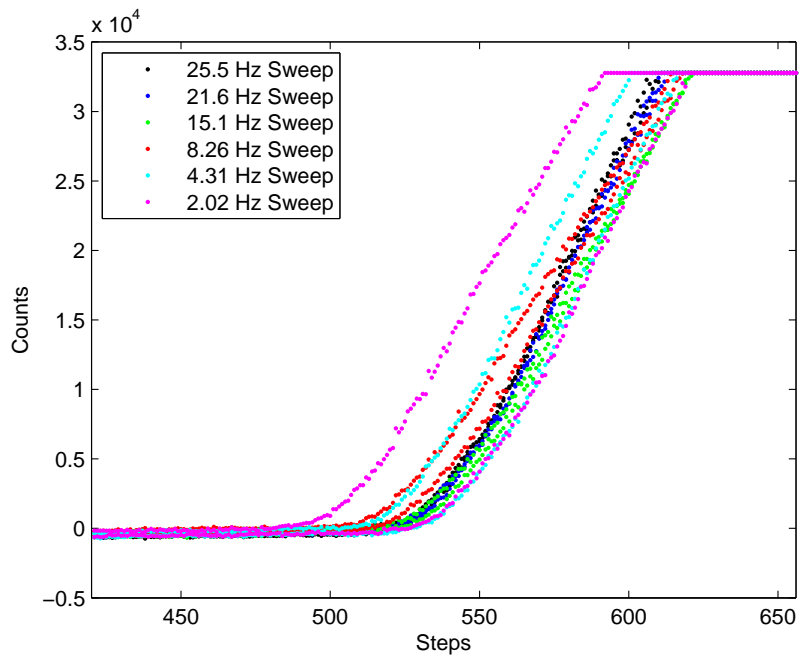Figure 3.11: Voltage Sweep Hysteresis

Figure 3.12: Voltage Sweep Hysteresis

Amatucci et al. (2001) have shown that heating a probe cleans the contamination and removes hysteresis. To clean the current needle probe, we need to heat it too. This can be done with an external heating source, or in the plasma chamber itself. Using a high density and high temperature plasma, most impurities can be removed, which is what is being done for the current plasma chamber probe. The probe used to create Figure 3.11 was cooked for almost 10 hours.

After another day of cooking in the plasmas chamber, data was taken and is shown in Figure 3.13. All hysteresis is gone from the electron saturation region as well as the retardation region. What this shows is that hysteresis caused by contamination can be removed by sweeping at higher frequencies.
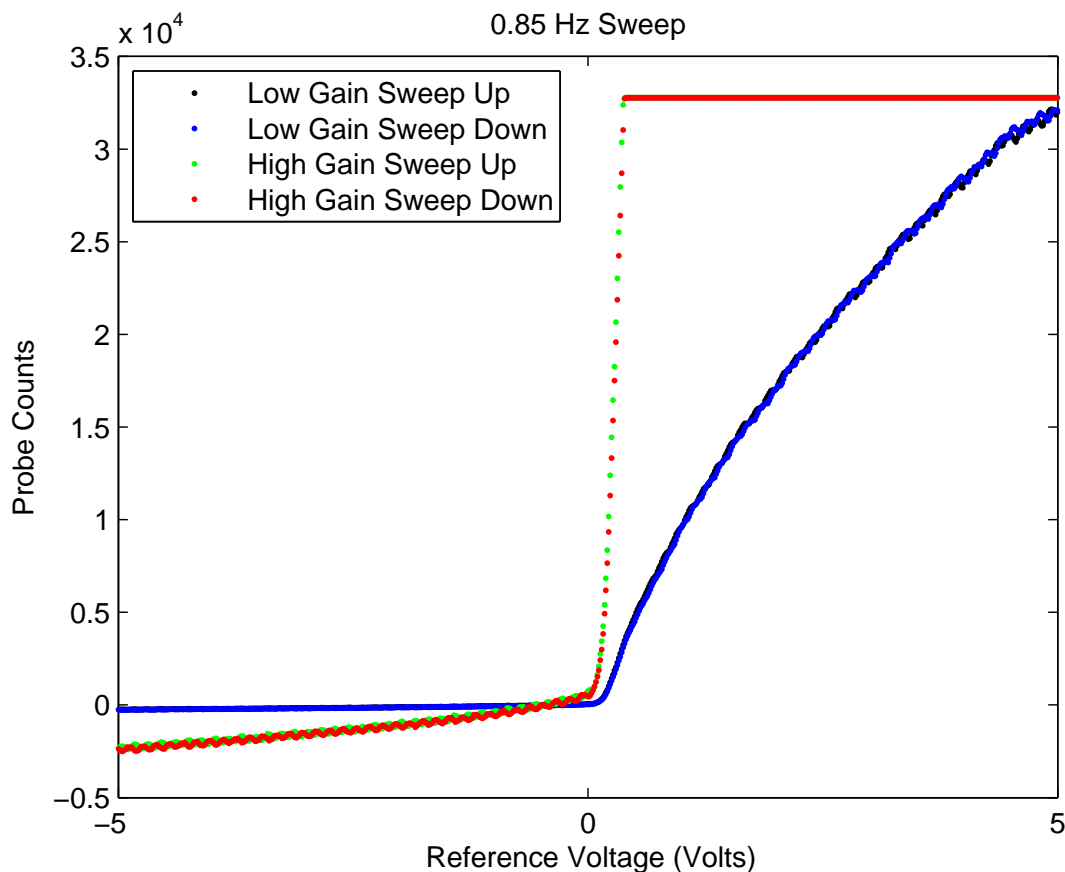

Figure 3.13: Voltage Sweep Hysteresis

3.3.3 Plasma Chamber SLP Results

The results for Figure 3.14 were taken with a very low filament current of 10.9 amps and a bias voltage of 9 volts. From this data, as shown in Figure 3.14, the electron density can be extracted from the electron saturation region. This is done by plotting the square of the probe current versus the reference voltage for the electron saturation region. A linear trendline is fitted to the data from which the slope is applied to equation 2.2. The temperature can be extracted from the retardation region using equation 1.5. The linear fit for the retardation current is applied to Figure 3.15. It is applied to the linear section right after the peak. The results for both regions are shown in table 3.4 and the linear trendlines are shown in following figures.
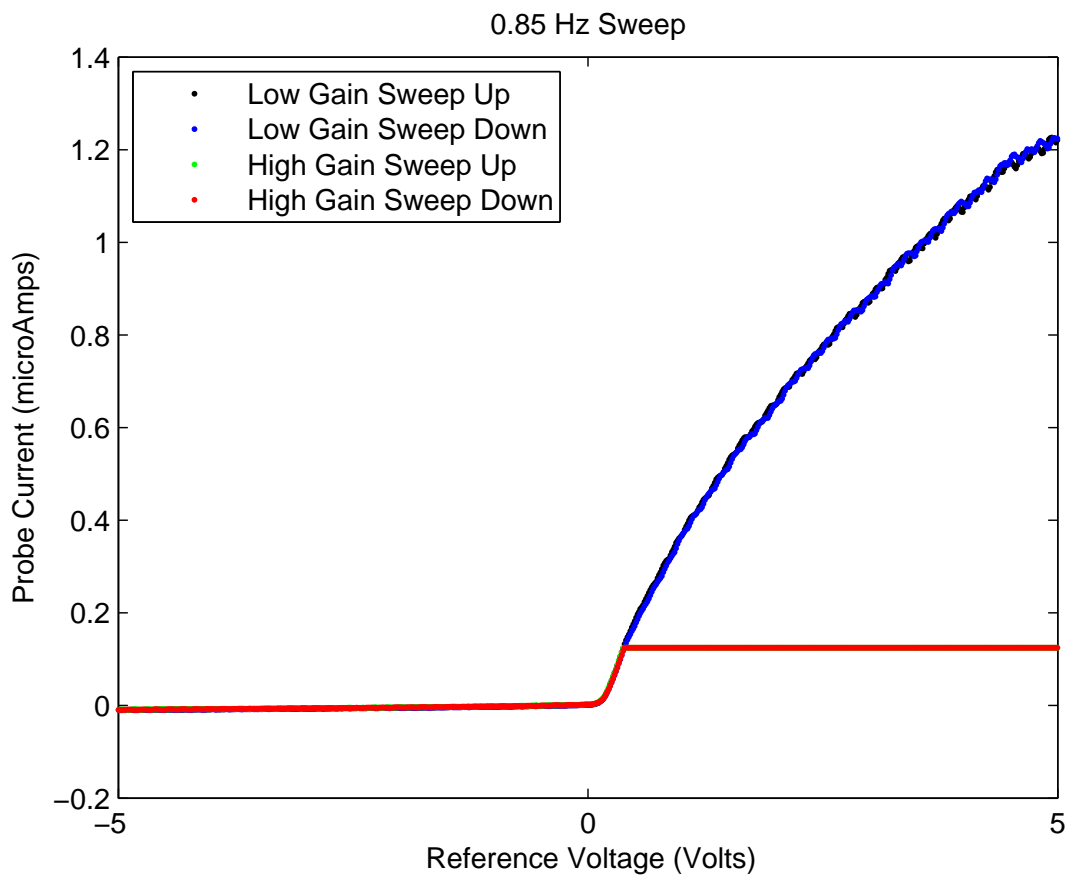
Figure 3.14: Voltage Sweep Hysteresis

Table 3.4: Plasma Chamber Results

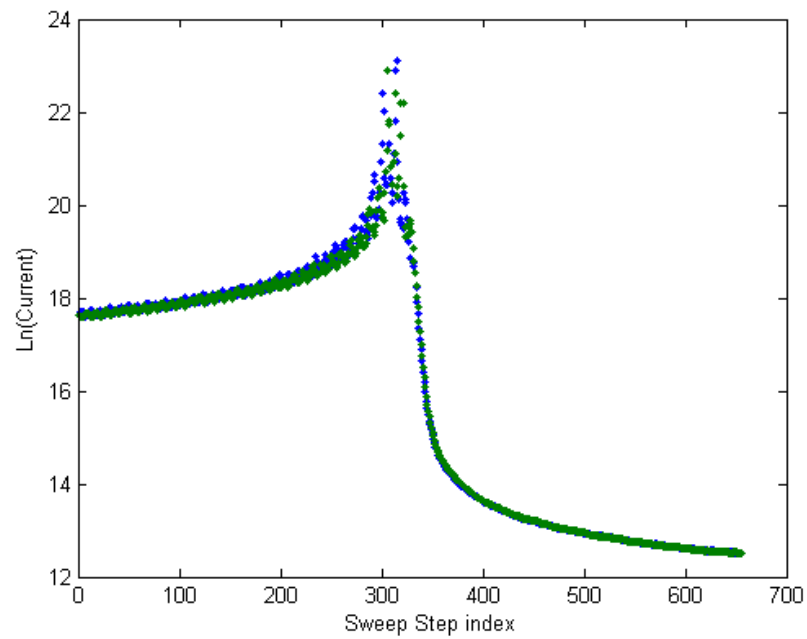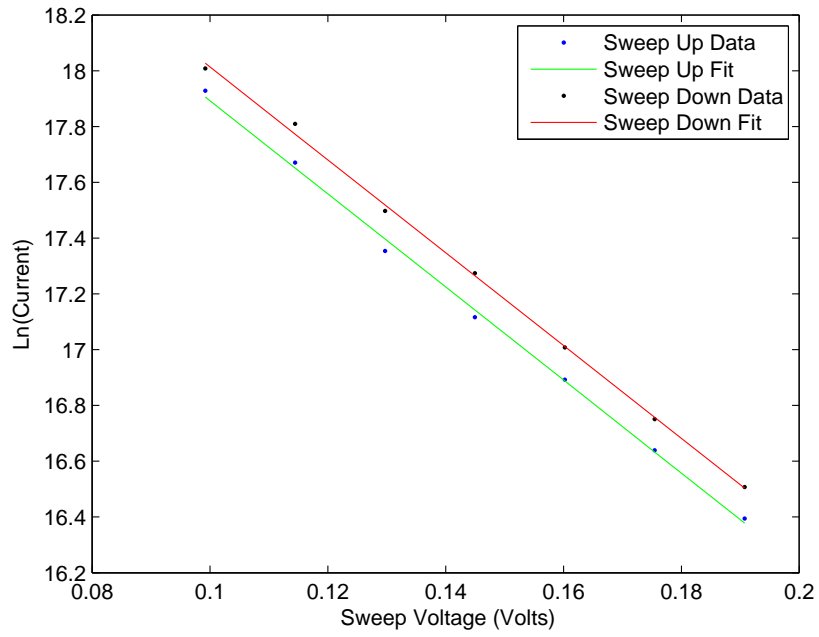|	Sweep Up	Sweep Down
Density	4.28×10^{10} electrons/m^3	4.34×10^{10} electrons/m^3
Temperature	695.96K	697.70K

Figure 3.15: Current for Linear Fitting

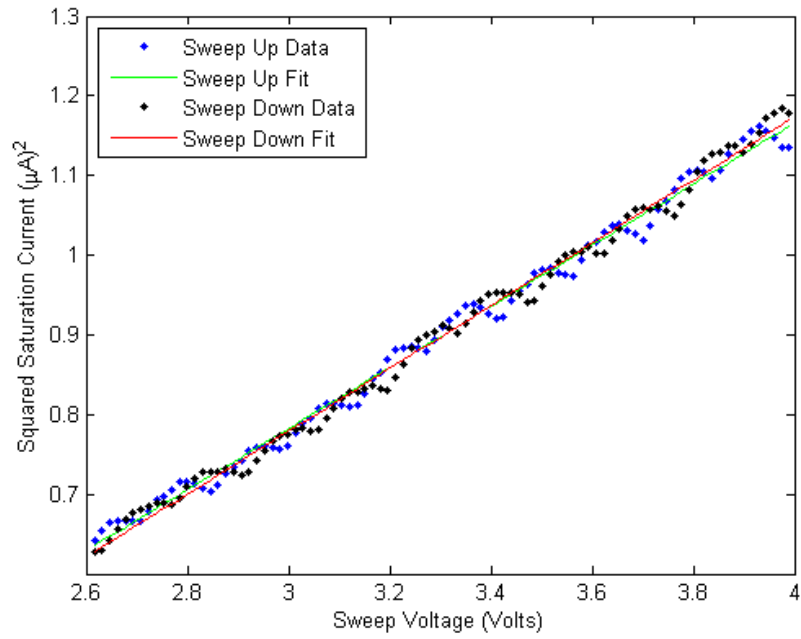Figure 3.16: Linear Fit for Electron Retardation Region

Figure 3.17: Linear Fit for Electron Saturation Region

Chapter 4

Summary and Future Work

The MTeX mission intends to study the turbulence associated with the mesospheric inversion layers in the altitude region of 75-110 km. The mission will launch two instrumented rockets from Poker Flat Research Range, Alaska in January 2015. The rockets will be launched within one hour of each other. The primary instrument of the mission is an ionization gauge supplied by the German team members. This ionization gauge will supply the neutral turbulence information. Embry-Riddle is responsible for the Langmuir probe suite of instruments. This suite consists of a sweeping Langmuir probe (SLP) and two separate implementations of fixed bias Langmuir probes: mDCP+ and mDCP-.

The SLP sweeps from -1.5 V to 3.5V, and will be used to derive absolute electron and ion density along with electron temperature and payload potential. The probe is a 4 cm long gold plated needle. The probe is deployed at the end of a 70 cm long boom in the fore of the payload to avoid making measurements in the rocket bow shock. The mDCP- (multi-needle probe) is a combination of three fixed-bias gold-plated 50 cm long needle probes. Each of the needles is biased at a unique potential: 3.3 V, 4 V and 5 V. This, in effect, gives three simultaneous samples along the electron saturation region curve. Using these three measurements, one can effectively derive high spatial resolution absolute electron density that is immune to low level of spacecraft charging. The mDCP+ is the most crucial instrument for the mission. It is made up of 3 spherical fixed bias probes, diameters of 0.75 inches, each with a

different metal electroplated on the sphere's surface. The three metals being used are Rhodium, Platinum and Indium. The objective is to measure the difference in the triboelectric currents collected by each sphere. The current difference is expected to be in the 10s of pA. Thus, it was crucial to design the probe electronics with very low noise. Using the information on triboelectric current difference, one can then diagnose the presence of mesospheric smoke particles along the rocket trajectory, and possibly even determine the primary metallic constituent of the smoke particles. Detecting the presence of smoke particles is crucial because their presence indicates that the turbulence in the plasma density cannot be taken as a proxy of neutral turbulence.

The work towards this thesis involved circuit design, layout, calibration and testing of all the above instruments. The detailed designs and code are presented as an appendix, as well as included on a CD as part of this thesis.

One of the crucial lessons learned on this project was the fact that the hand soldering of TQFP parts is just too error-prone and costs a lot of time. Thus, in future, all such designs that involve quad flat package chips should be outsourced to firms capable of populating the boards using automated machining techniques.

It is recommended that logarithmic amplifiers are incorporated into two gain channels. That way, a more dynamic range will be available for the instrument to operate in. The only additional feature the payloads require is a connector for external programming. This way the payloads can be programmed without opening them. This would allow for programming while the payloads are integrated with the rocket.

Bibliography

Amatucci, W. E., P. W. Schuck, D. N. Walker, P. M. Kintner, S. Powell, B. Holbeck, and D. Leonhardt (2001), Contamination-free sounding rocket langmuir probe, *Reviews of Scientific Instruments*, *72*(4), 2052.

AnalogDevices (2012), Ad7656-1: 250 ksps, 6-channel, simultaneous sampling, bipolar 16-bit adc, *Tech. rep.*

AtmelCorporation (2012), At32uc3c series complete, *Tech. rep.*

Barjatya, A. (2007), Langmuir probe measurements in the ionosphere, Ph.D. thesis, Utah State University, DigitalCommons@USU.

Bekkeng, T. A. (2009), Prototype development of a multi-needle langmuir probe system, Master's thesis, University of Oslo Department of Physics.

Blix, T., E. Thrane, and O. Andreassen (1990), In situ measurements of the fine-scale structure and turbulence in the mesosphere and lower thermosphere by means of electrostatic positive ion probes, *J. Geophys. Res.*, *95*, 5533–5548.

Brace, L. H. (1998), Langmuir probe measurements in the ionosphere, in *Measurement Techniques in Space Plasmas: Particles, Geophysical Monograph Series*, vol. 102, edited by R. F. P. et al., pp. 23–35, American Geophysical Union, Washington, D.C.

Cabrera, N., and N. F. Mott (1949), Theory of the oxidation of metals, *Reports on Progress in Physics*, *12*, 163–184.

Cho, J. Y. N. (1992), On the role of charged aerosols in polar summer mesospheric echoes, *J. Geophys. Res.*, *97*, 875.

Croskey, C., J. Mitchell, R. Goldberg, T. Blix, M. Rapp, R. Latteck, M. Friedrich, , and B. Smiley (2004), Coordinated investigation of plasma and neutral density fluctuations and particles, *Geophys. Res. Lett.*, *31*, doi:10.1029/2004GL020169.

Croskey, C. L., J. D. Mitchell, M. Friedrich, F. J. Schmidlin, , and R. A. Goldberg (2006), In-situ electron and ion measurements and observed gravity wave effects in the polar mesosphere during the macwave program, *Annales Geophysicae*, *24*, 1167–1278.

Ferguson, D. C., B. Hillard, and T. L. Morton (2001), The floating potential probe (fpp) for iss: Operations and initial results, in *7th Spacecraft Charging Technology Conference at ESTEC, Noordwijk, The Netherlands, ESA SP-476*, edited by R. A. Harris, p. 3655.

Goldberg, R. A., et al. (2004), The macwave/midas rocket and groundbased measurements of polar summer dynamics: Overview and mean state structure, *Geophys. Res. Lett.*, *31*.

Goldberg, R. A., et al. (2006), The macwave program to studygravity wave influences on the polar mesosphere, *Annales Geophysicae*, *24*, 1159–1173.

Harper, W. R. (1967), *Contact and Frictional Electrification, ser. Monographs on the Physics and Chemistry of Materials*, Oxford University Press.

Kelley, M., D. Farley, and J. Rottger (1987), The effect of cluster ions on anomalous vhf backscatter from the summer polar mesosphere, *Geophys. Res. Lett.*, *14*, 1031–1034.

LinearTechnology (2009), Application note 124, *Tech. rep.*

LinearTechnology (2014), Ltc6655 - precision buffered reference, *Tech. rep.*

Lubken, F., M. Rapp, T. Blix, and E. Thrane (1998), Microphysical and turbulent measurements of the schmidt number in the vicinity of polar mesosphere summer echoes, *Geophys. Res. Lett.*, *25*, 893–896.

Lucky, Z., W. Gekelman, and P. Pribyl (2014), *Manual for Embry Unmagnetized Plasma Device*, Z. Lucky enterprise, 1 ed.

McNeil, W. J., E. Murad, and J. M. C. Plane (2002), *Models of meteoric metals in the atmosphere*, 265-287 pp., Cambridge University Press, Cambridge.

Mott-Smith, H. M., and I. Langmuir (1924), Studies of electric discharges in gas at low pressures, p. 616.

Mott-Smith, H. M., and I. Langmuir (1926), The theory of collectors in gaseous discharges, *Phys. Rev.*, *28*, 727.

Oyama, K.-I. (1975), A systematic investigation of several phenomena associated with contaminated langmuir probes, *Planetary Space Sciences*, *24*(12), 183–190, institute of Space and Aeronautical Sciences, University of Tokyo.

Plane, J. M. C. (2004), A new time-resolved model of the mesospheric na layer: constraints on the meteor input function, *Atmos. Chem. Phys. Discuss.*, *4*, 39–69.

Rapp, M., B. Strelnikov, S. Wilms, F. Lubken, J. Gumbel, and H. Henkel (2003a), A new detector for the in situ measurement of meteoric dust particles in the middle atmosphere, in *Proceedings of the 16th ESA symposiumon European Rocket and Balloon Programmes and Related Research*, edited by E. B. Warmbeina, pp. 379–384, St. Gallen, Switzerland, eSA SP-530.

Rapp, M., F. Lubken, P. Hoffman, G. Baumgarten, and T. Blix (2003b), Pmse dependence on aerosol charge number density and aerosol size, *J. Geophys. Res.*, *108*, 8441, doi:10.1029/2002JD002650.

Sternovsky, Z., M. Horanyi, and S. Robertson (2001), Charging of dust particles on surfaces, *Journal of Vacuum Science and Technology*, *19*(5), 2533–2541.

Strelnikova, I., and M. Rapp (2009), Measurements of meteor smoke particles during the ecoma-2006 campaign: 1. particle detection by active photoionization, *J. Atmos. Sol. Terr. Phys.*, *71*, 477–485.

Szuszczewicz, E. P. (1972), Area influences and floating potentials in langmuir probe measurements, *Journal of Applied Physics,*, *43*(3), 874.

TexasInstrument (2009), Dac8581: 16-bit, high speed, low noise, voltage output digital to analog converter, *Tech. rep.*

TexasInstruments (2007), Ina2128: Dual, low power instrumentation amplifier, *Tech. rep.*
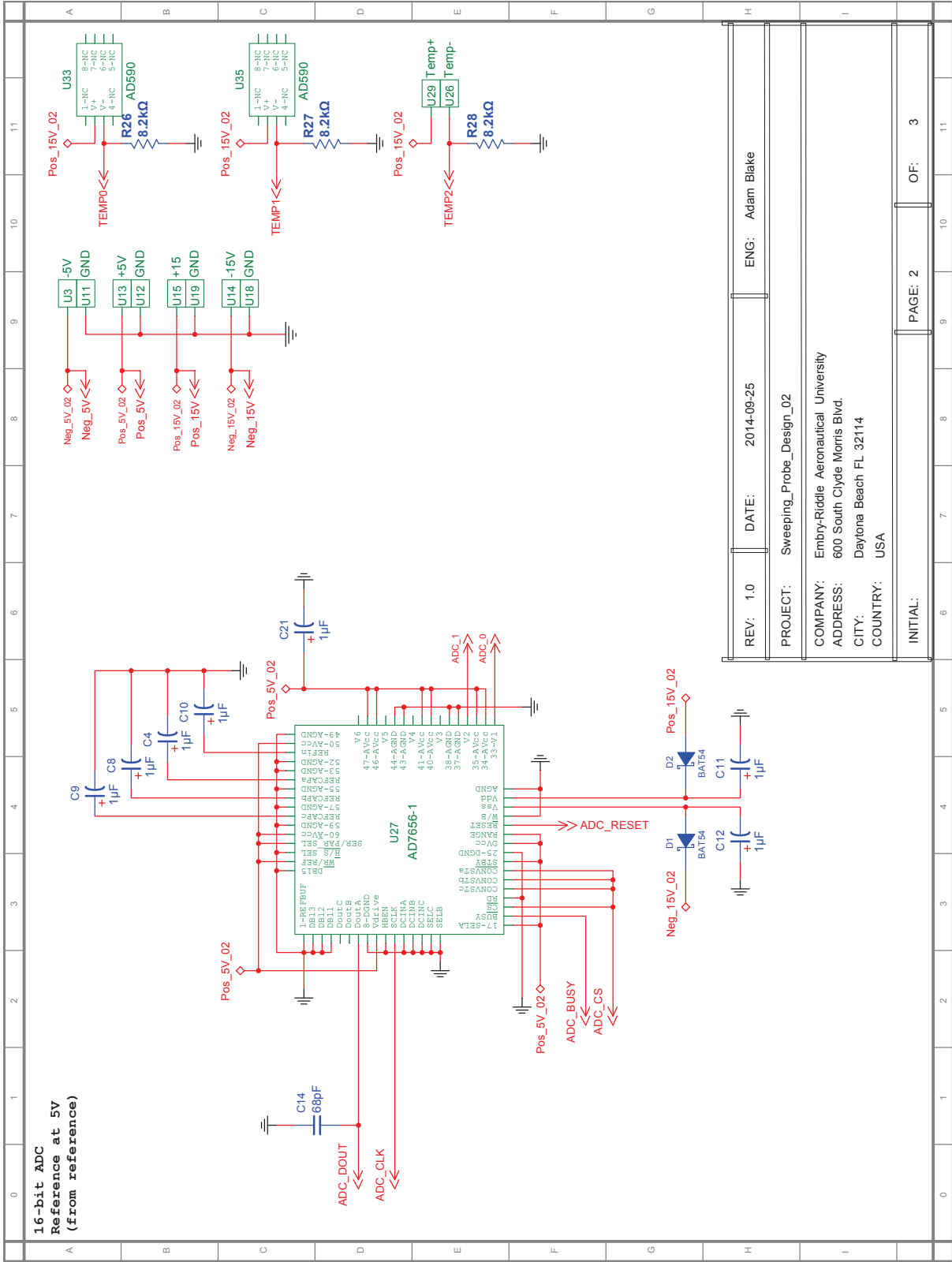
Chapter 5

Appendix

5.1   Circuit Design

5.1.1 Sweeping Langmuir Probe Board

16-bit ADC
Reference at 5V
(from reference)
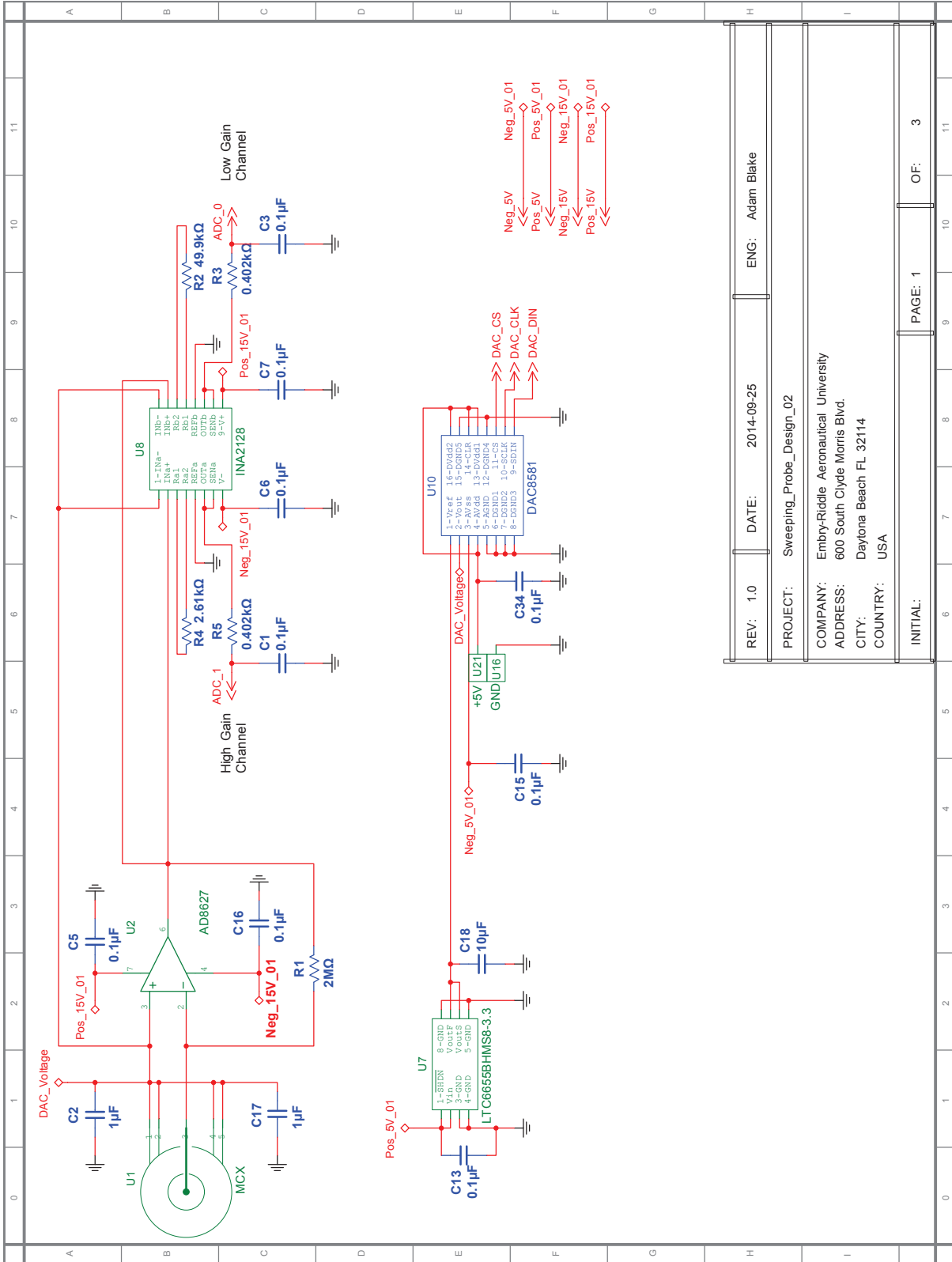
ENG: Adam Blake

DATE: 2014-09-25

REV: 1.0

PROJECT: Sweeping_Probe_Design_02

COMPANY: Embry-Riddle Aeronautical University
ADDRESS: 600 South Clyde Morris Blvd.
CITY: Daytona Beach FL 32114
COUNTRY: USA

INITIAL:

PAGE: 2   OF: 3

5.1.2 Fixed Langmuir Probe Board mDCP+

16-bit ADC
Reference at 5V
(from reference)

5.1.3 Fixed Langmuir Probe Board mDCP-

16-bit ADC
Reference at 5V
(from reference)

5.1.4 PowerBoard
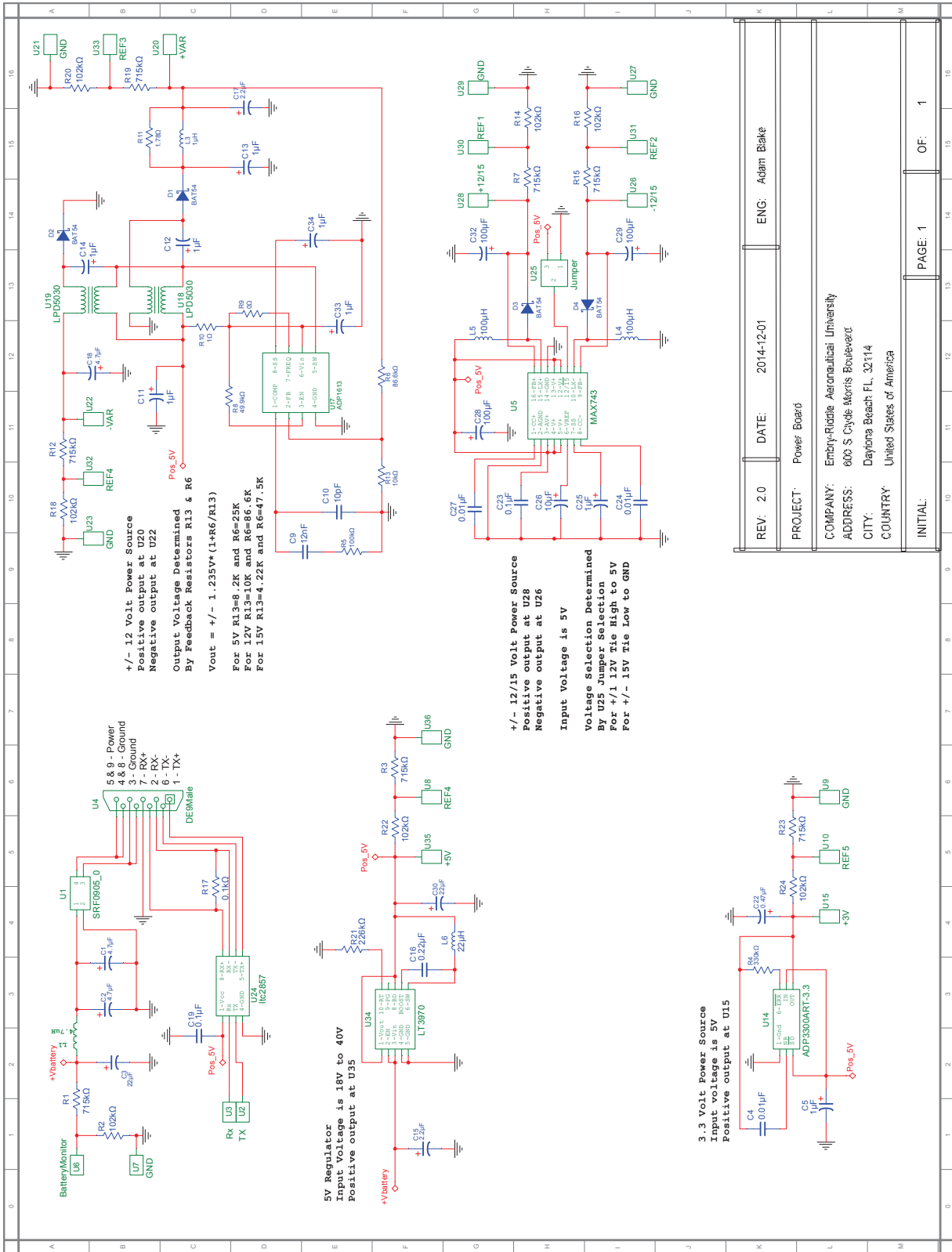
5.2 Printed Circuit Board Layout

5.2.1 SLP Layout

SLP Board Outline
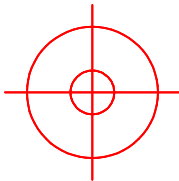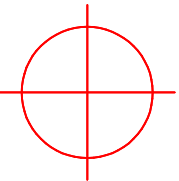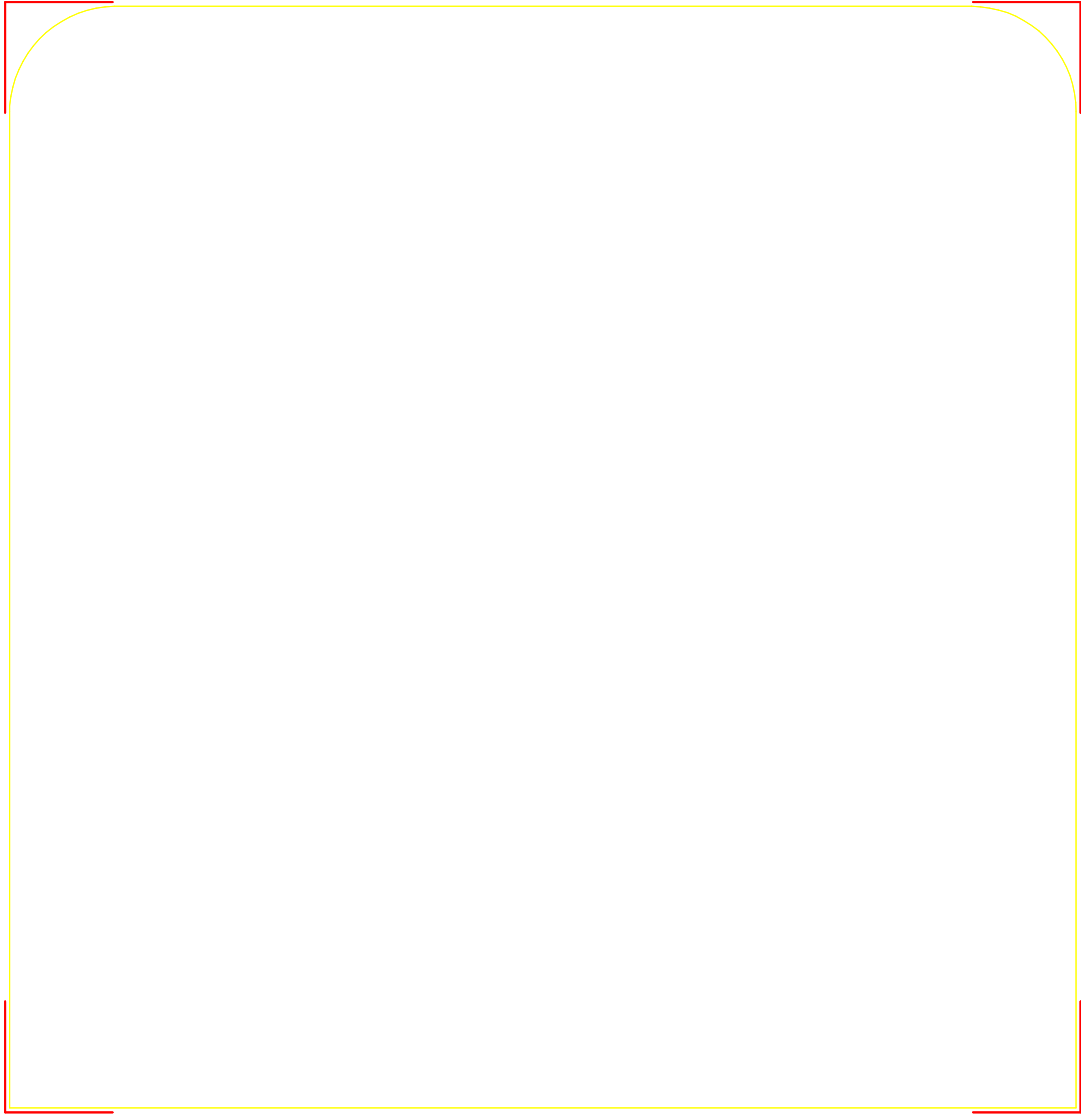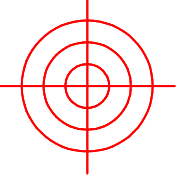
SLP Top Silkscreen
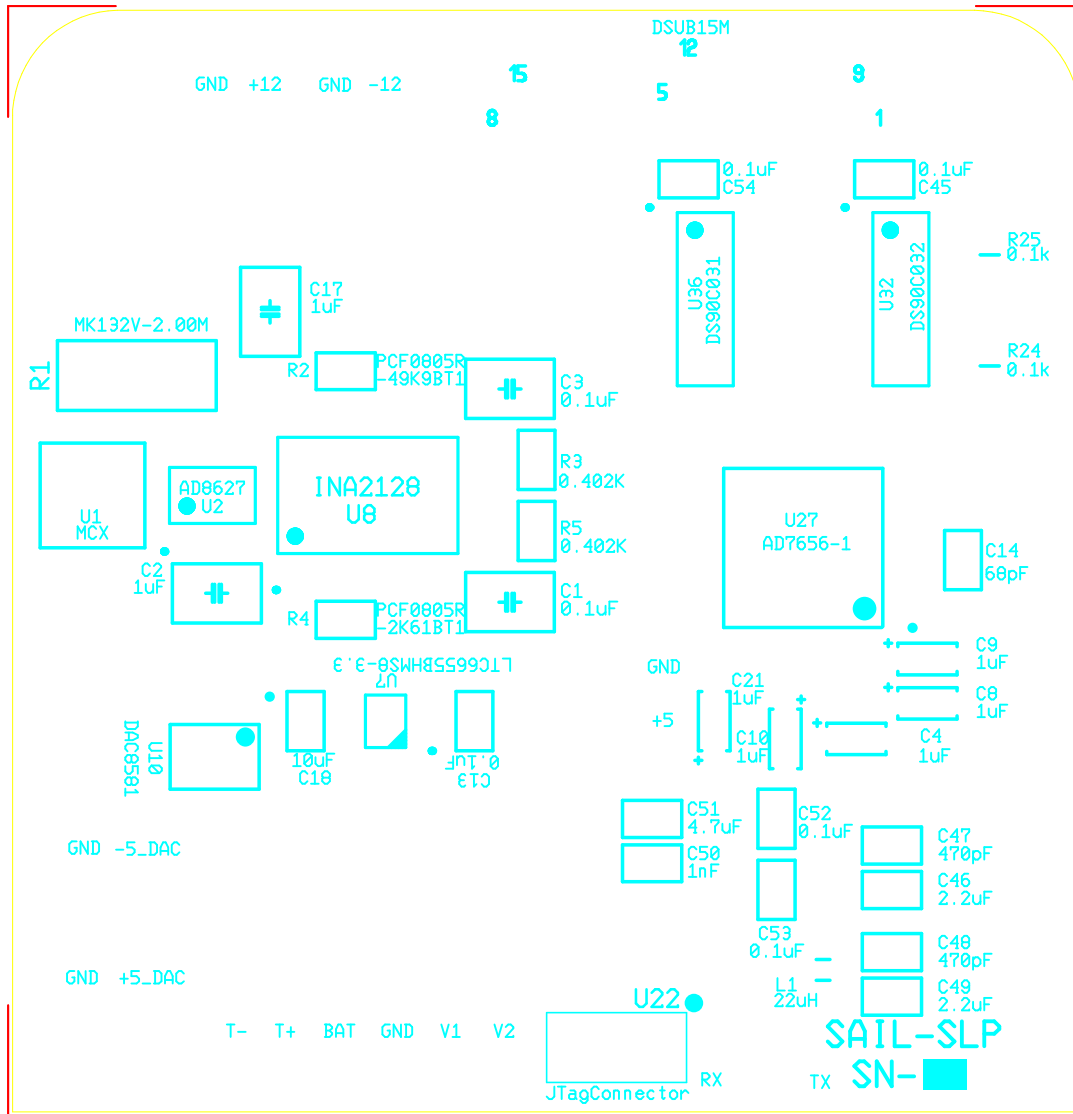
SLP Top Copper
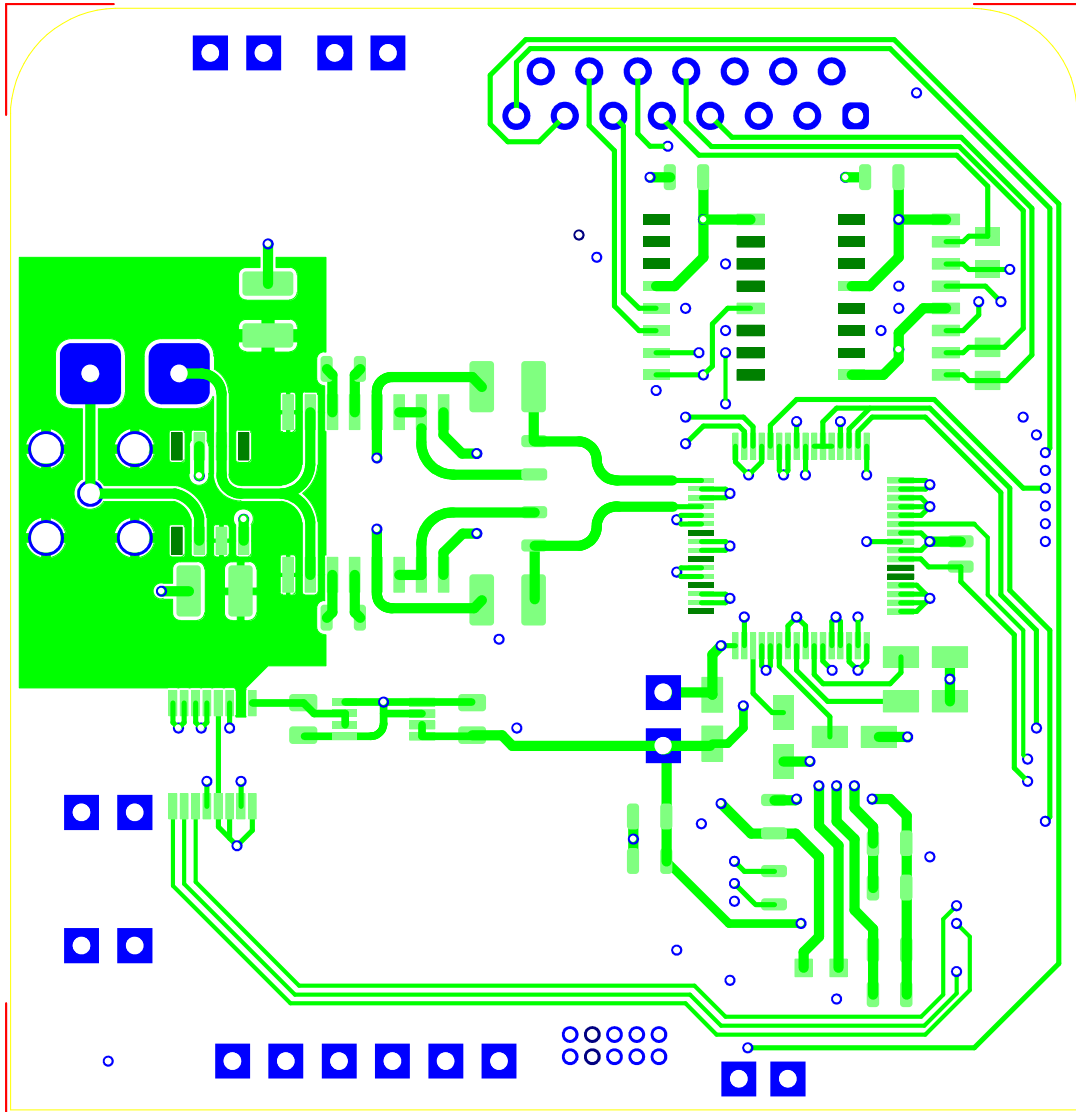
SLP Inner Copper Layer 1
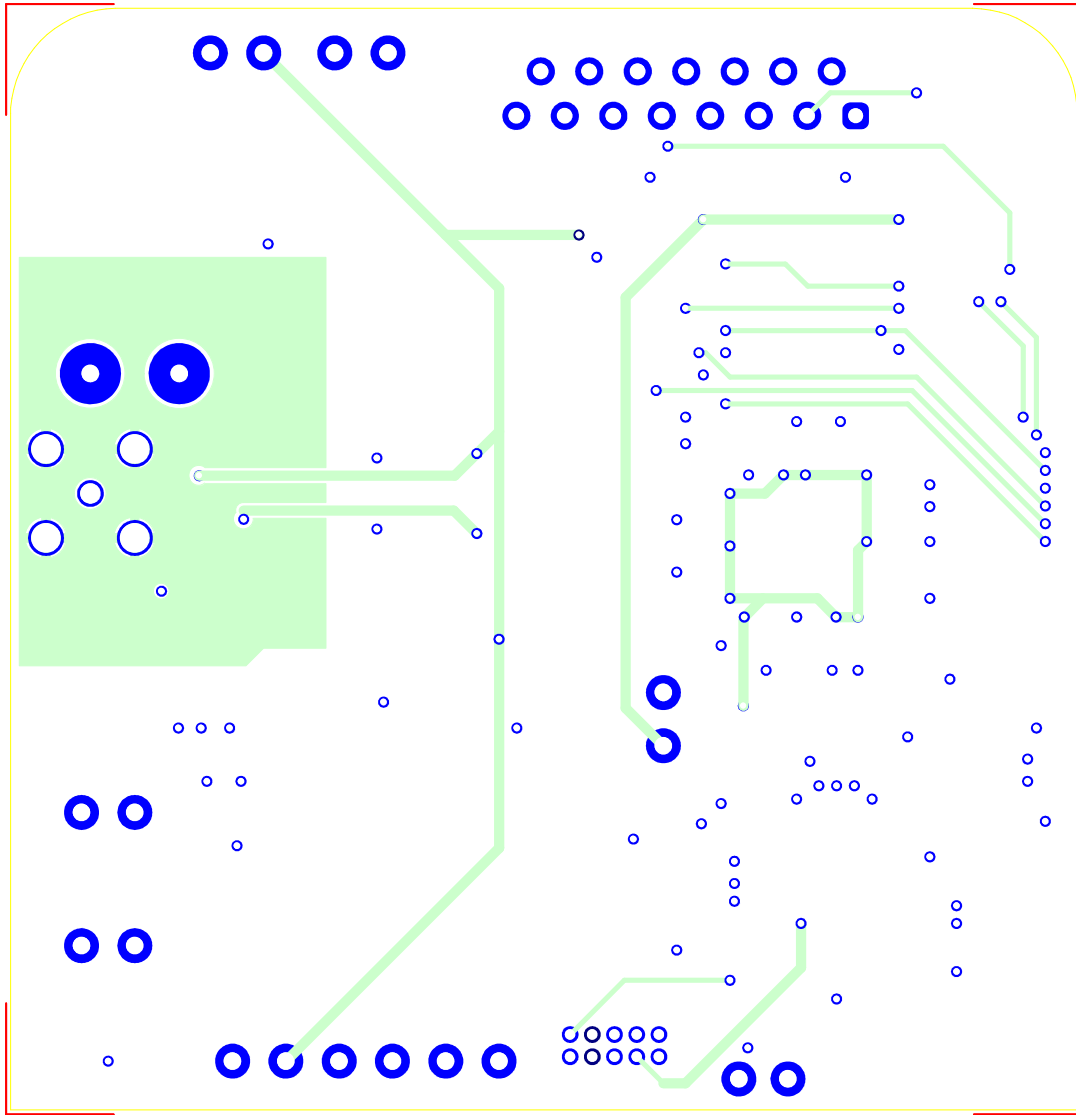
SLP Inner Copper Layer 2

SLP Bottom Copper
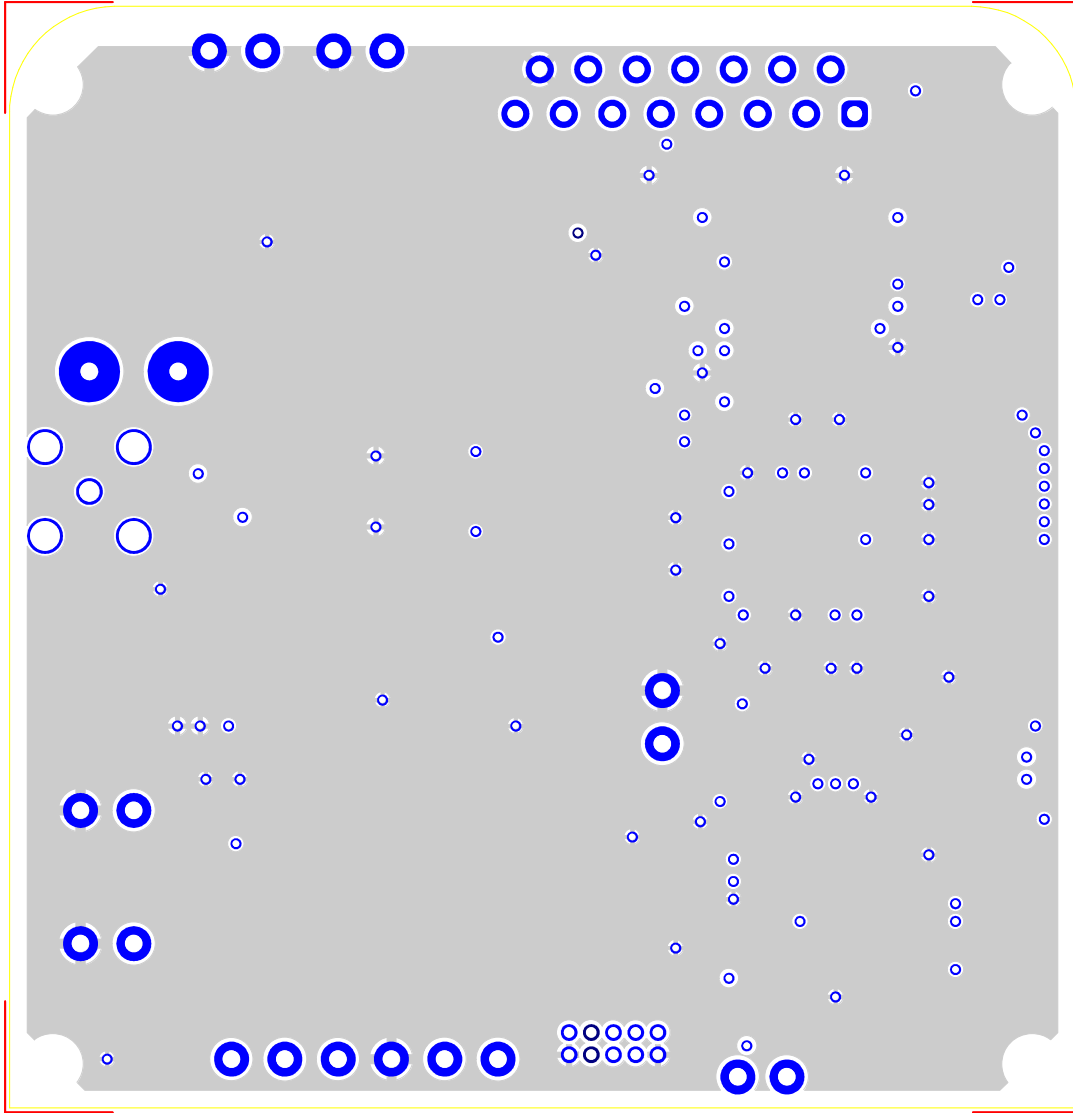
SLP Bottom Silkscreen
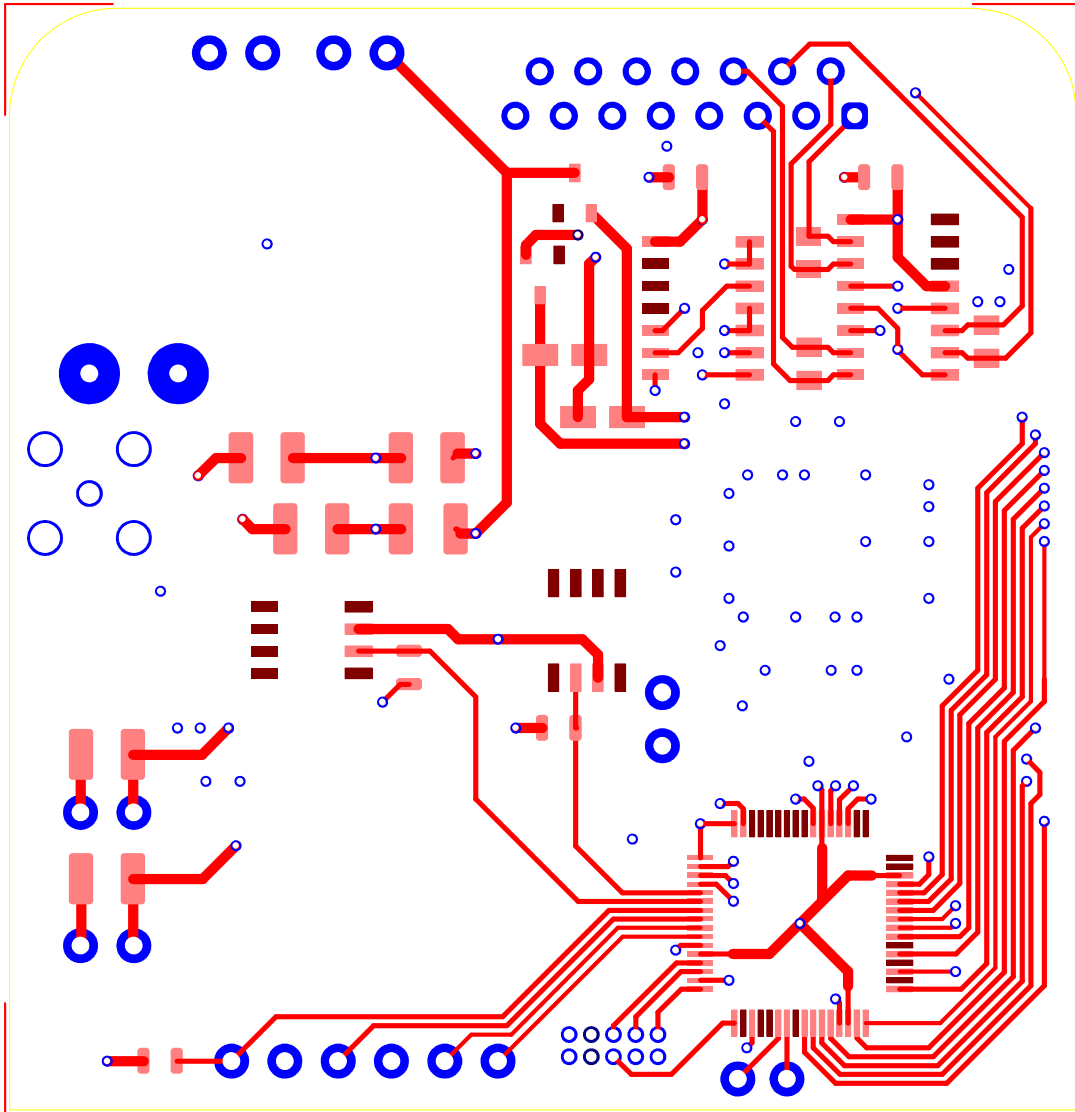
5.2.2   mDCP+ Layout

mDCP+ Board Outline
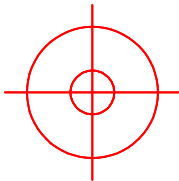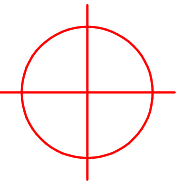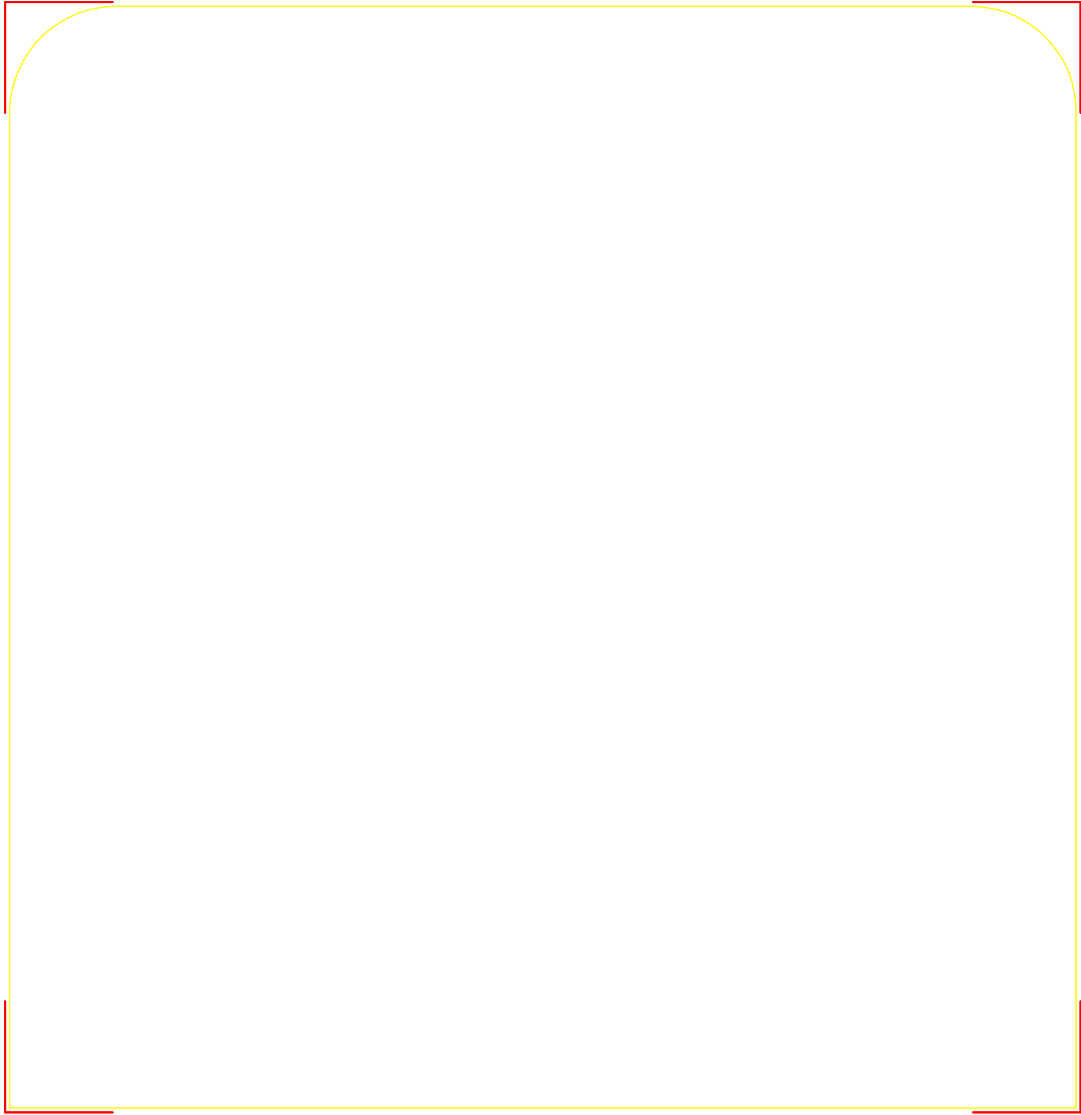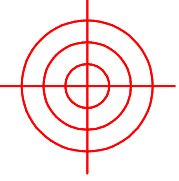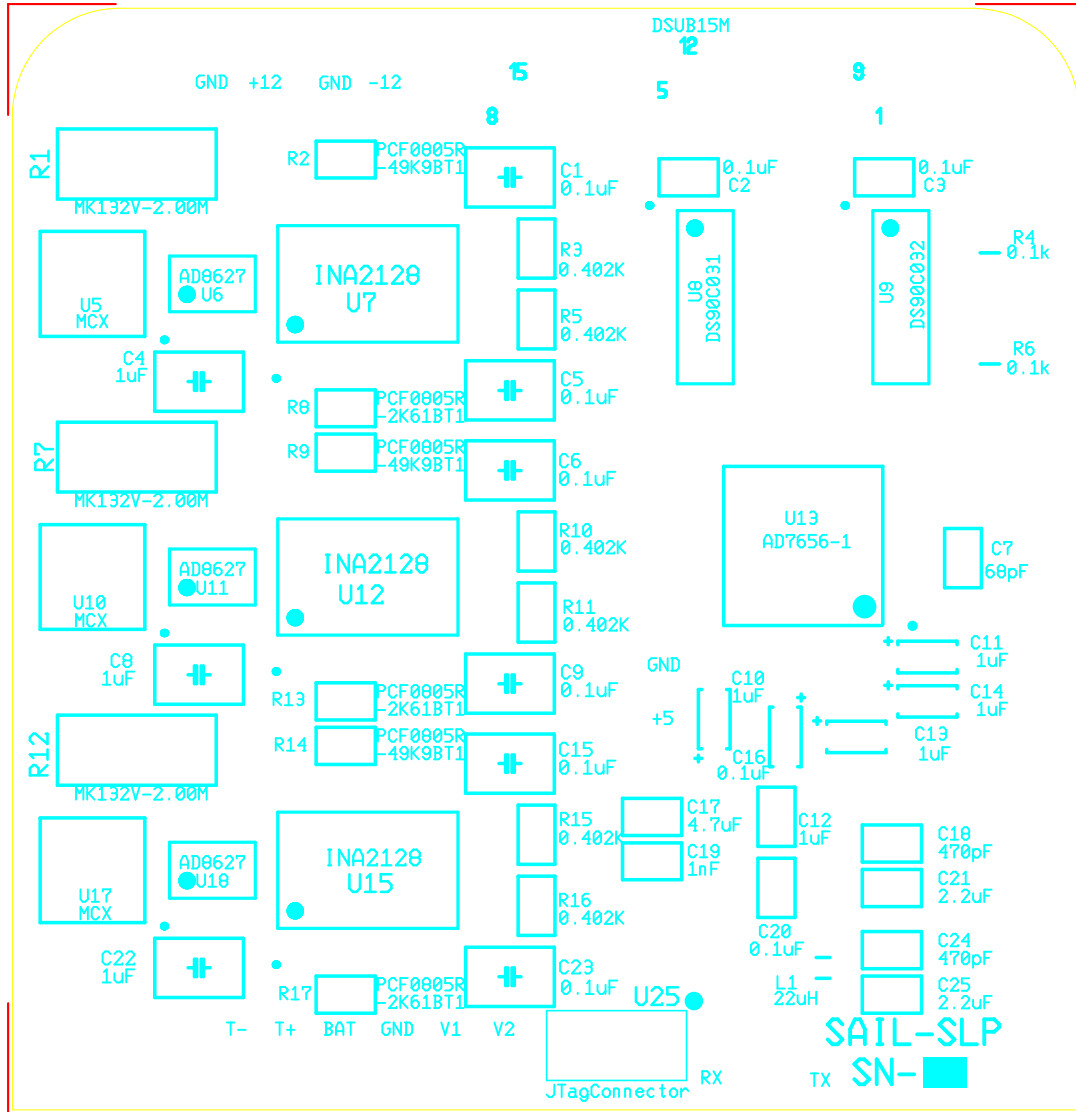
mDCP+ Top Silkscreen

mDCP+ Top Copper

mDCP+ Inner Copper Layer 1
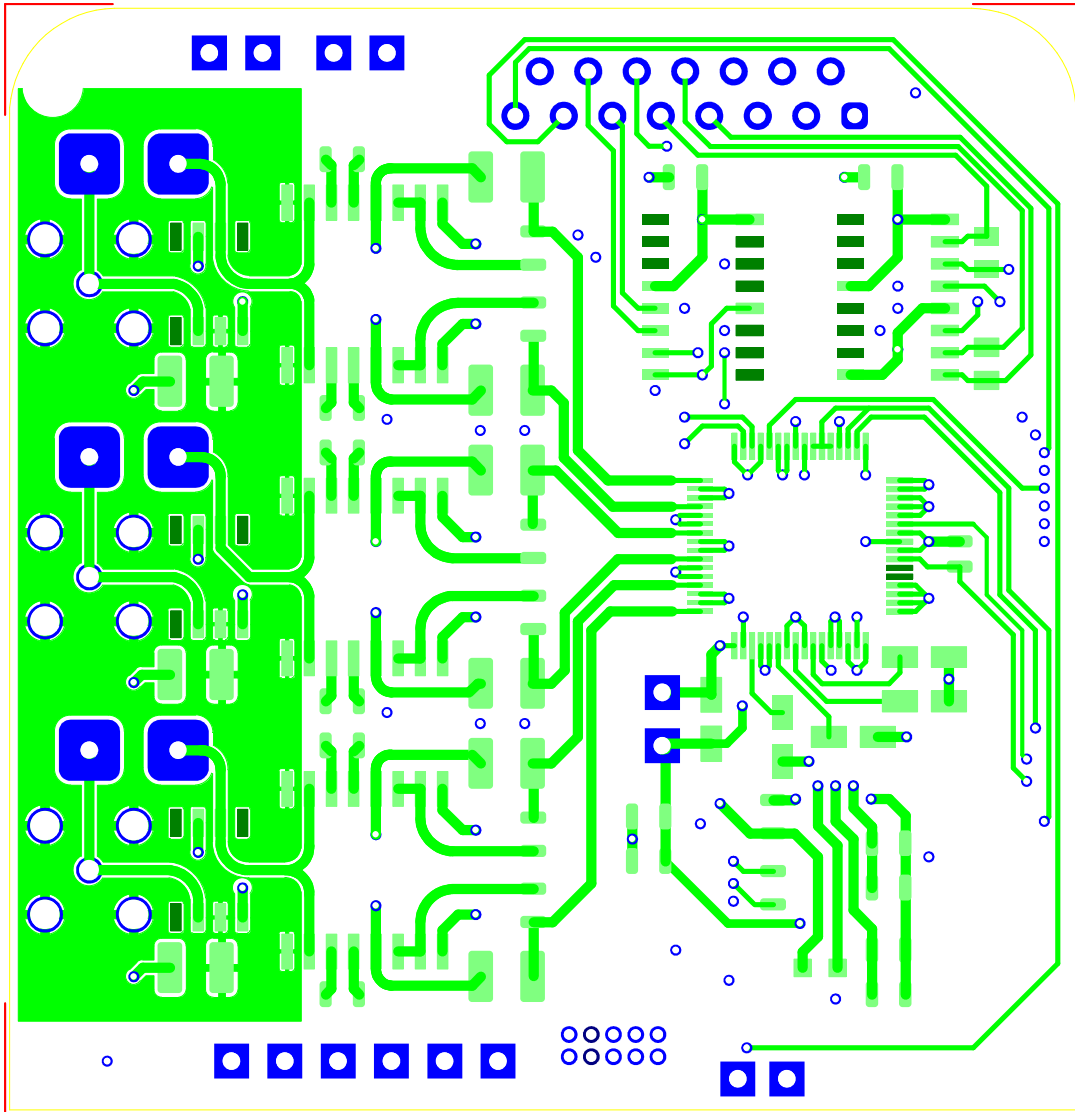
mDCP+ Inner Copper Layer 2
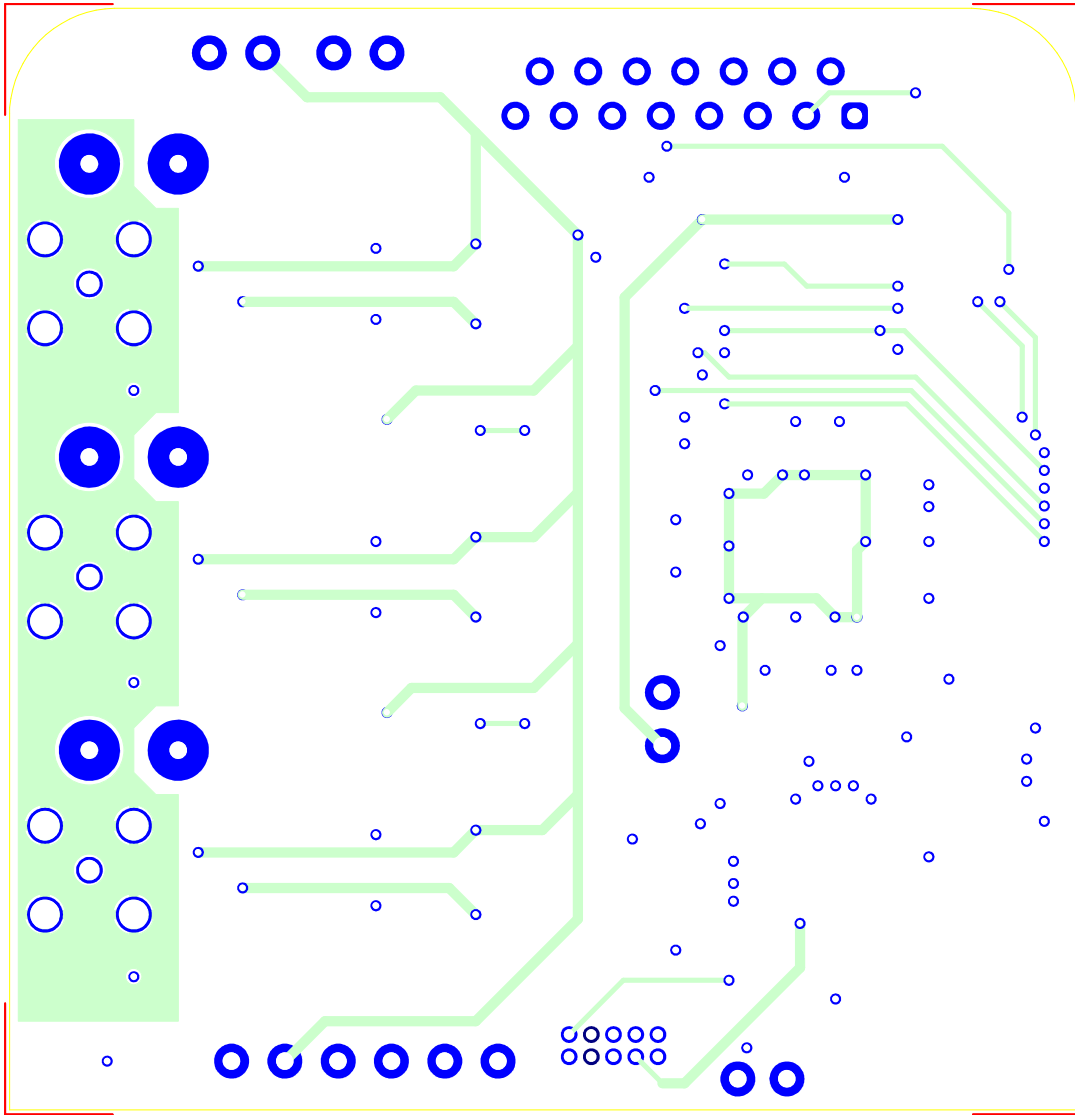
mDCP+ Bottom Copper
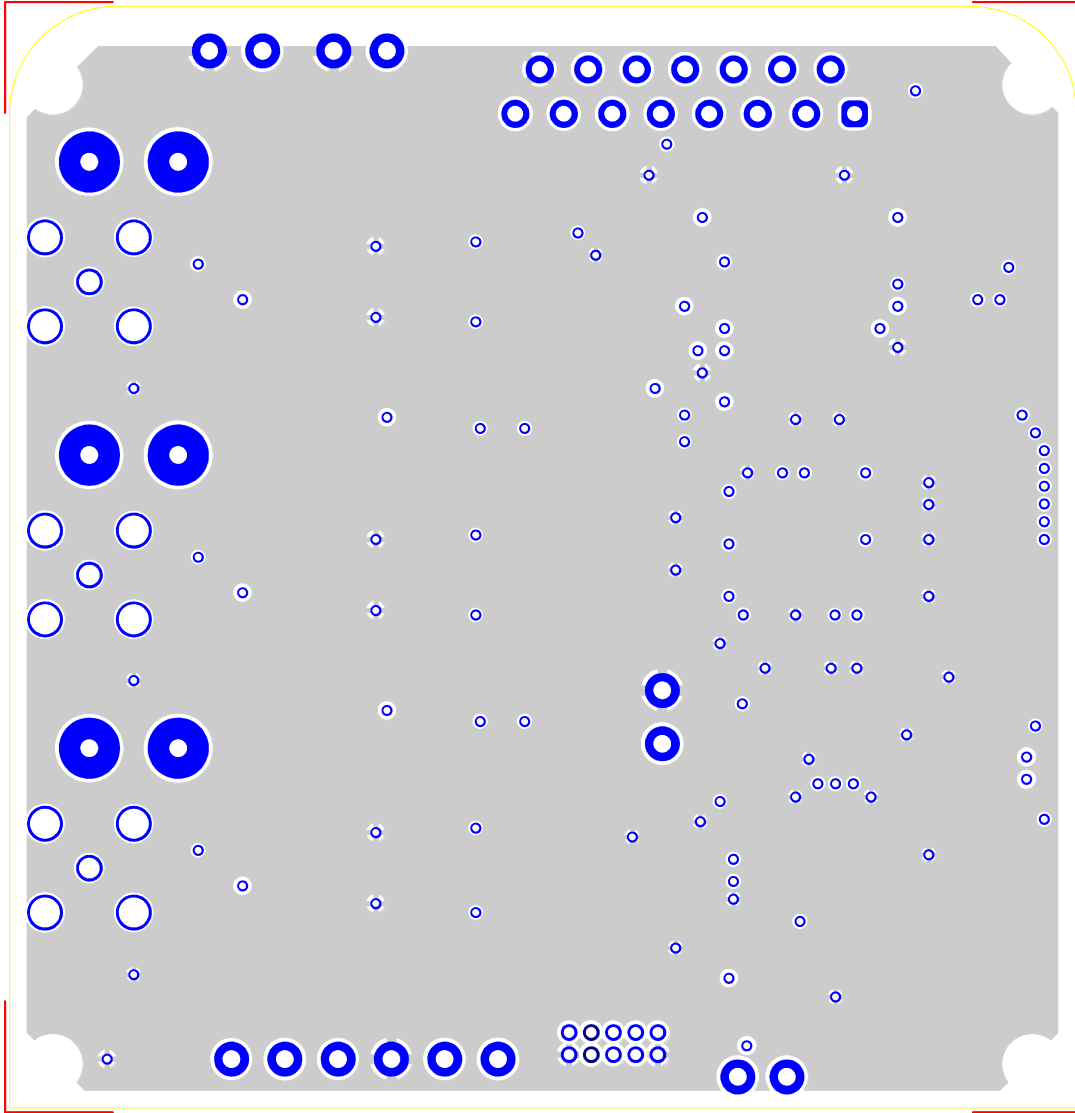
mDCP+ Bottom Silkscreen
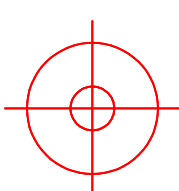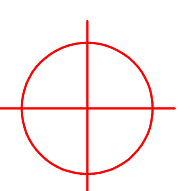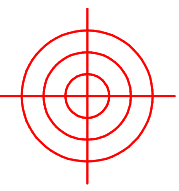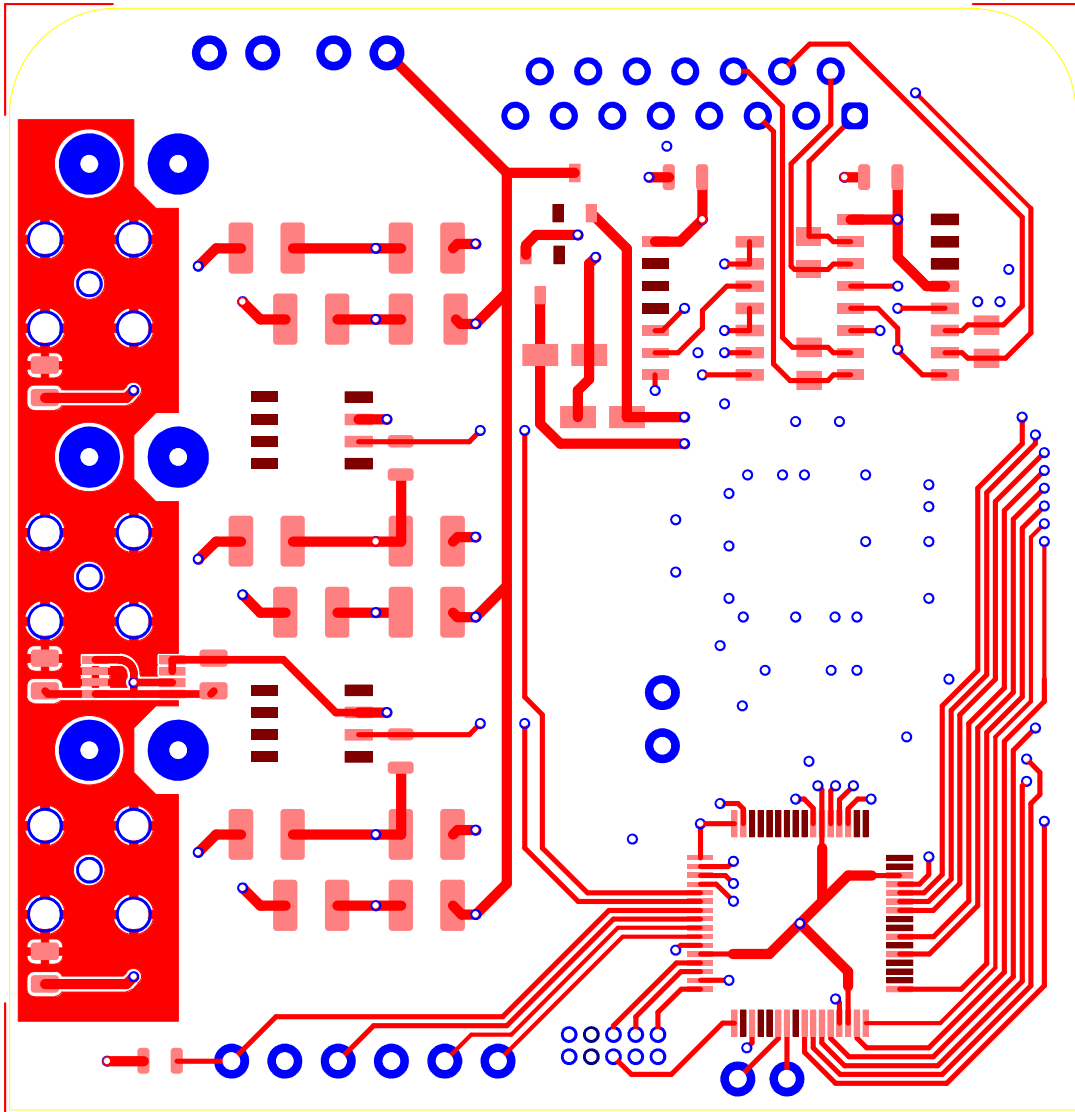
5.2.3 mDCP- Layout

mDCP- Board Outline
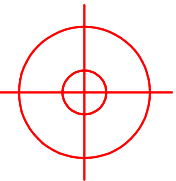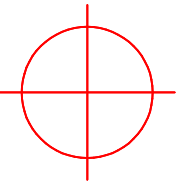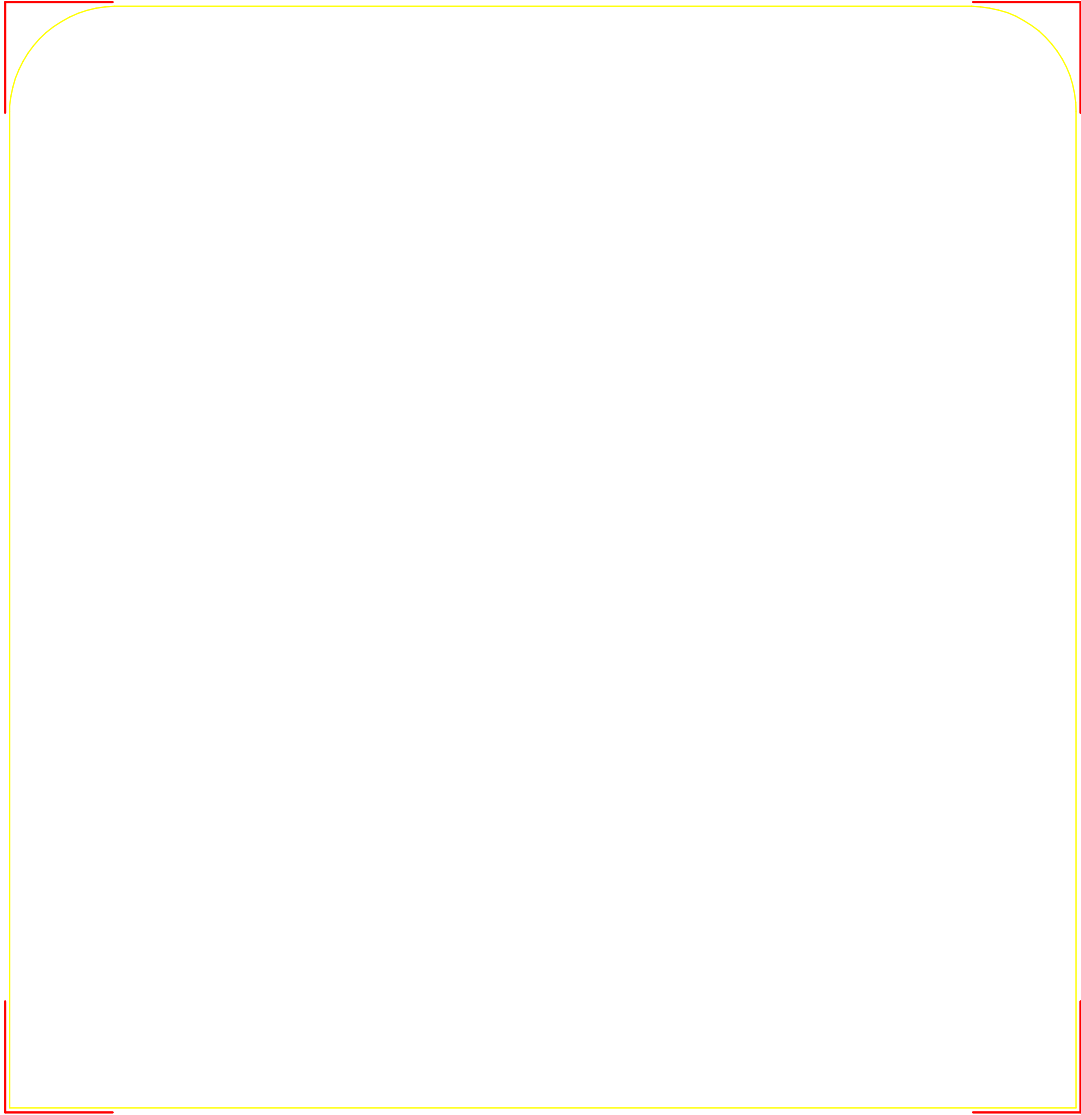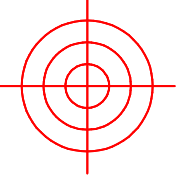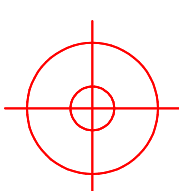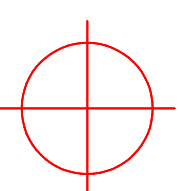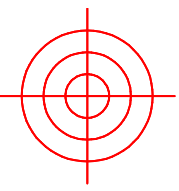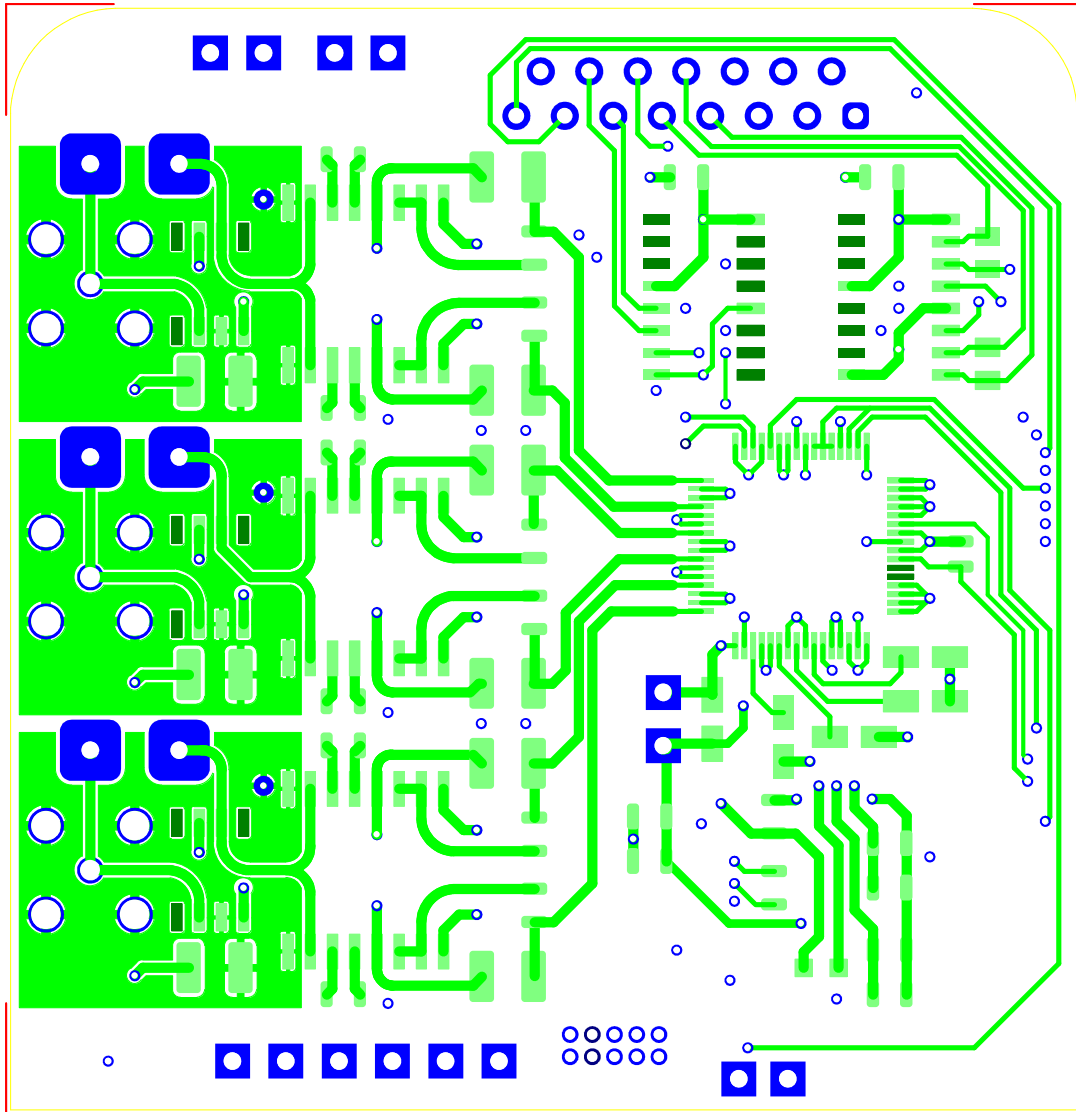
mDCP- Top Silkscreen

mDCP- Top Copper

mDCP- Inner Copper Layer 1
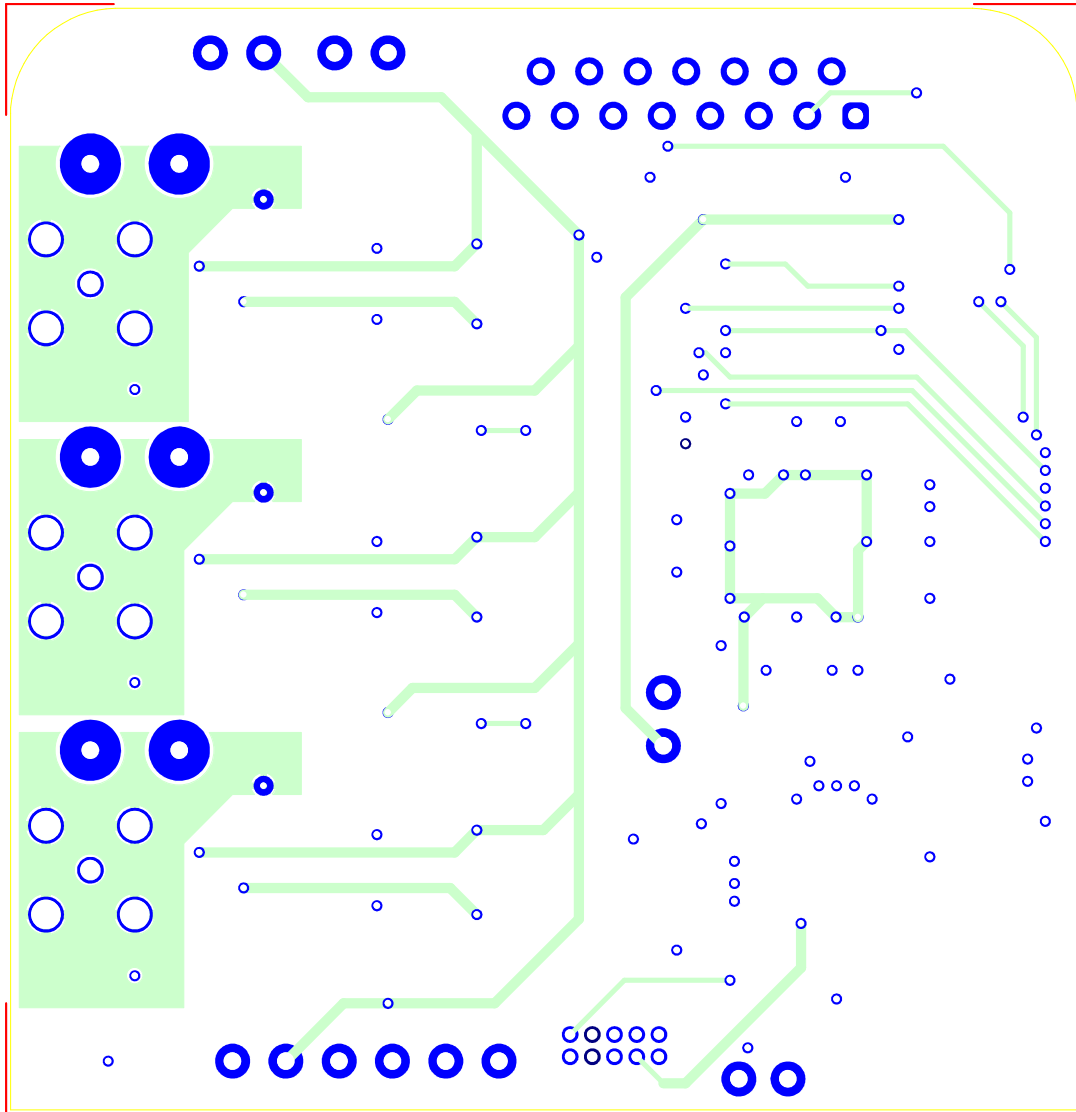
mDCP- Inner Copper Layer 2
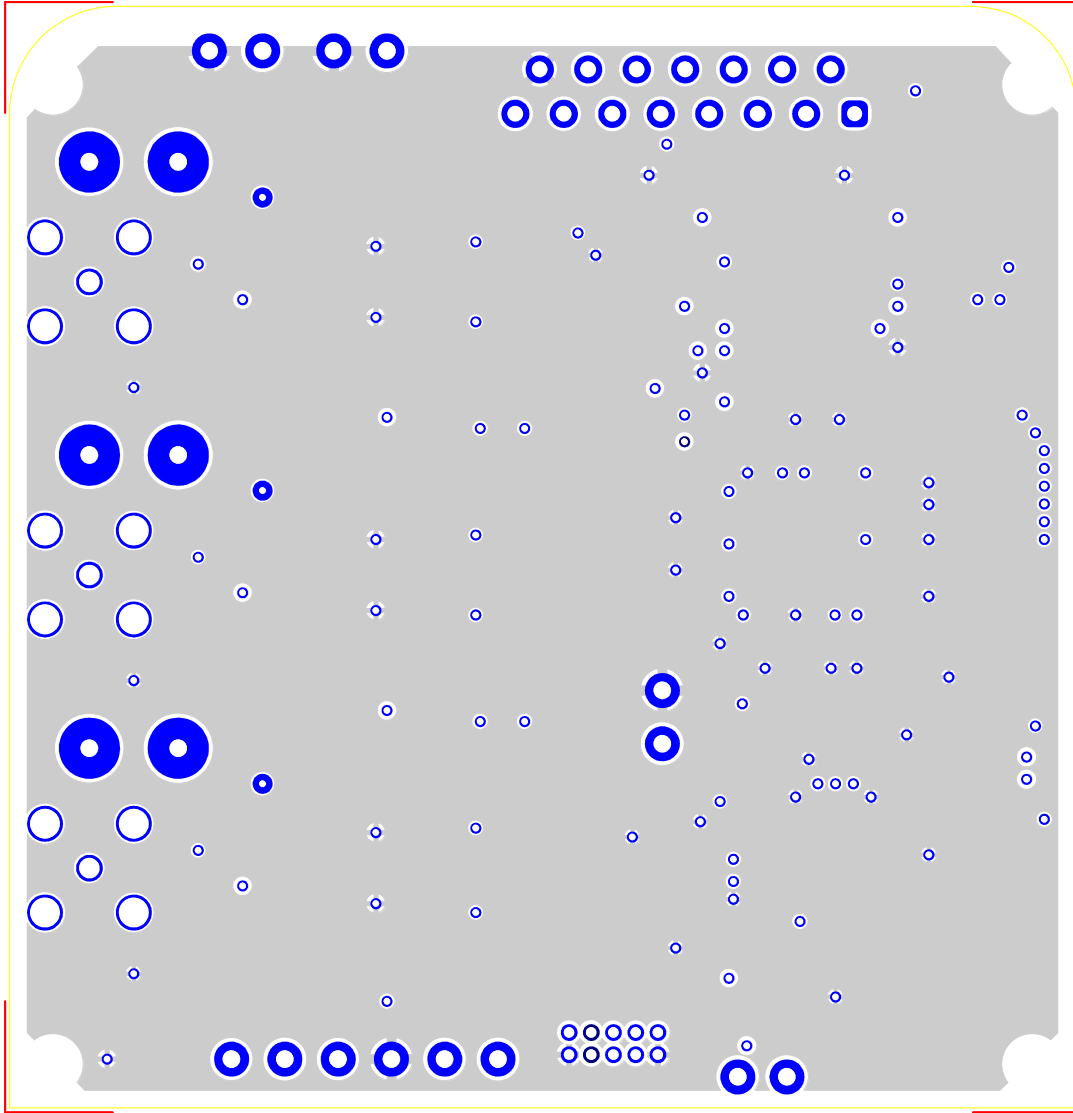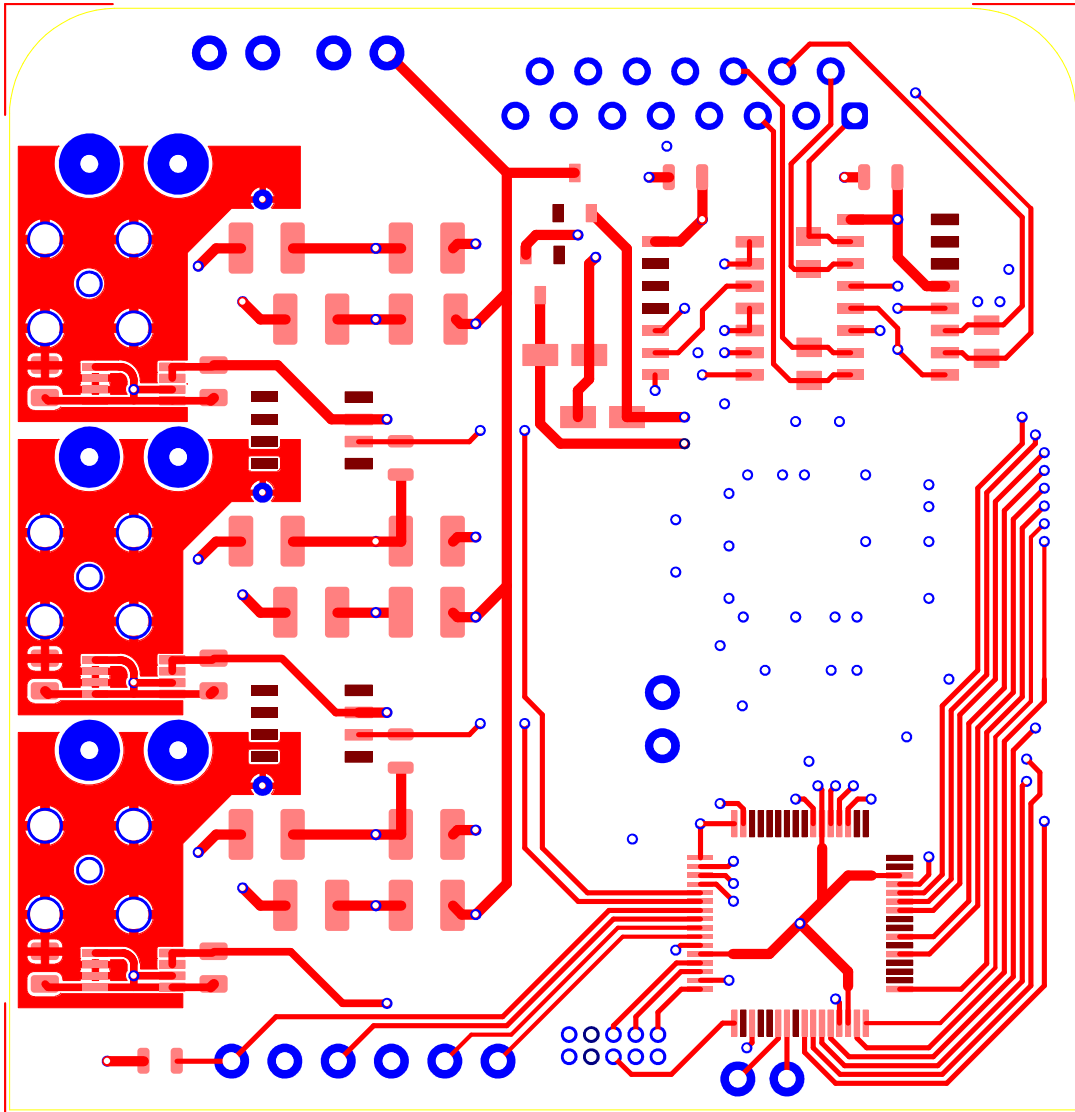
mDCP- Bottom Copper

mDCP- Bottom Silkscreen
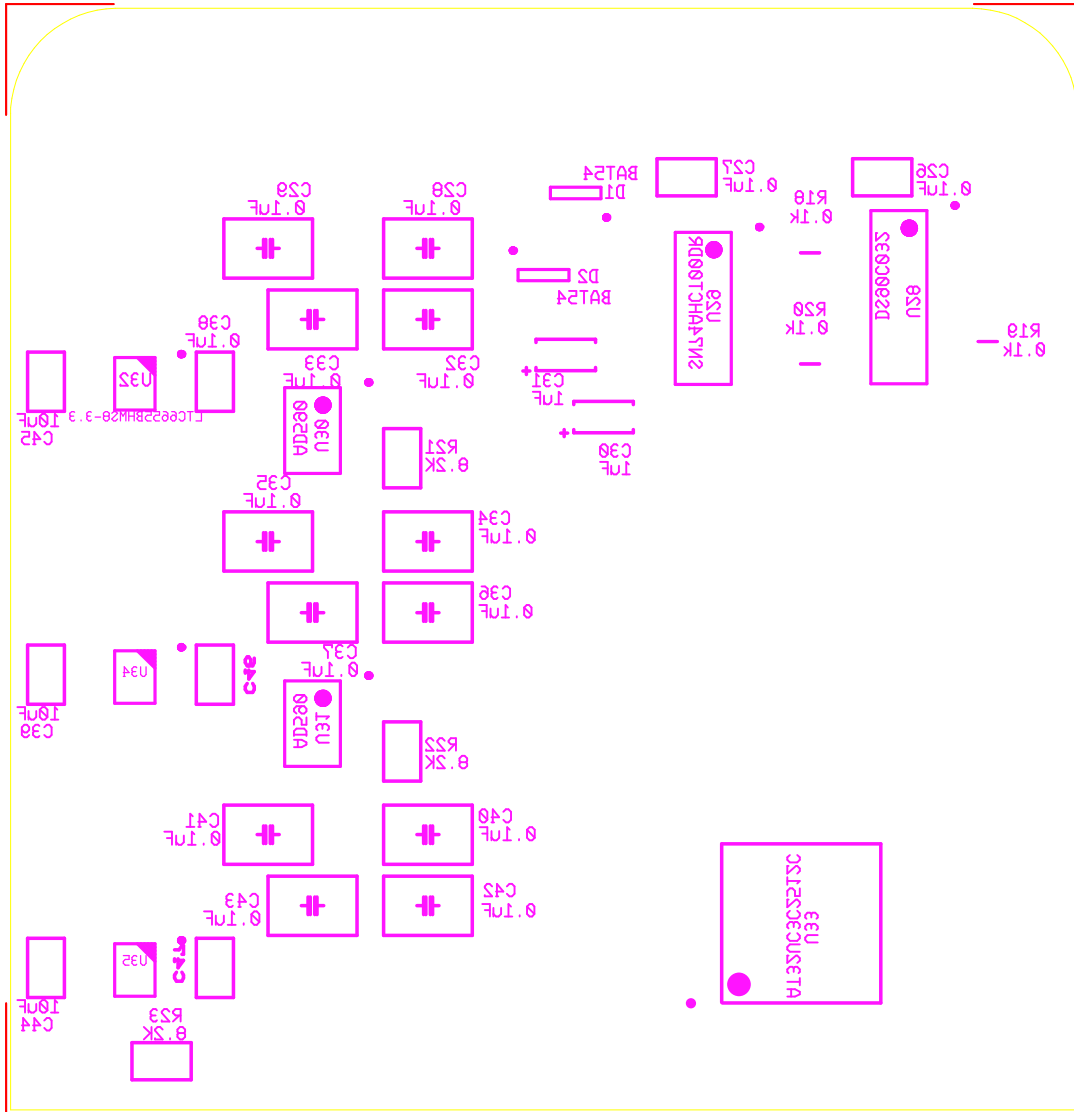
5.2.4   Power Board Layout

Power Board Outline
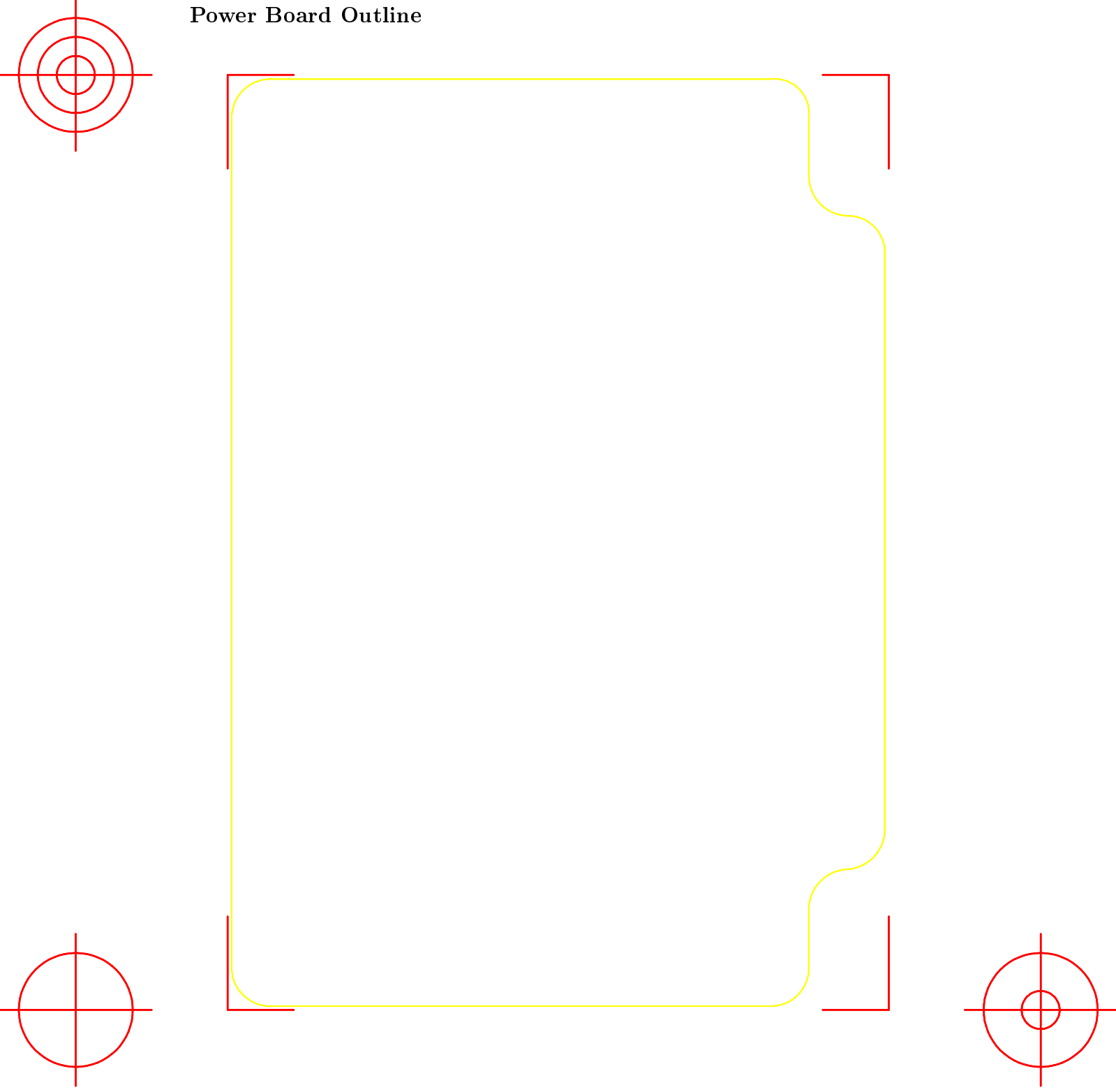
Power Top Silkscreen
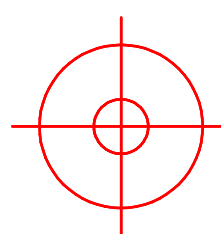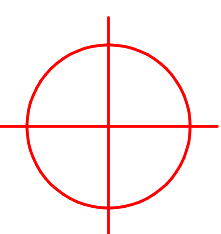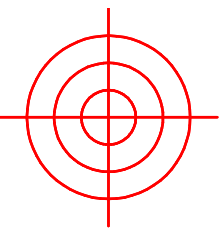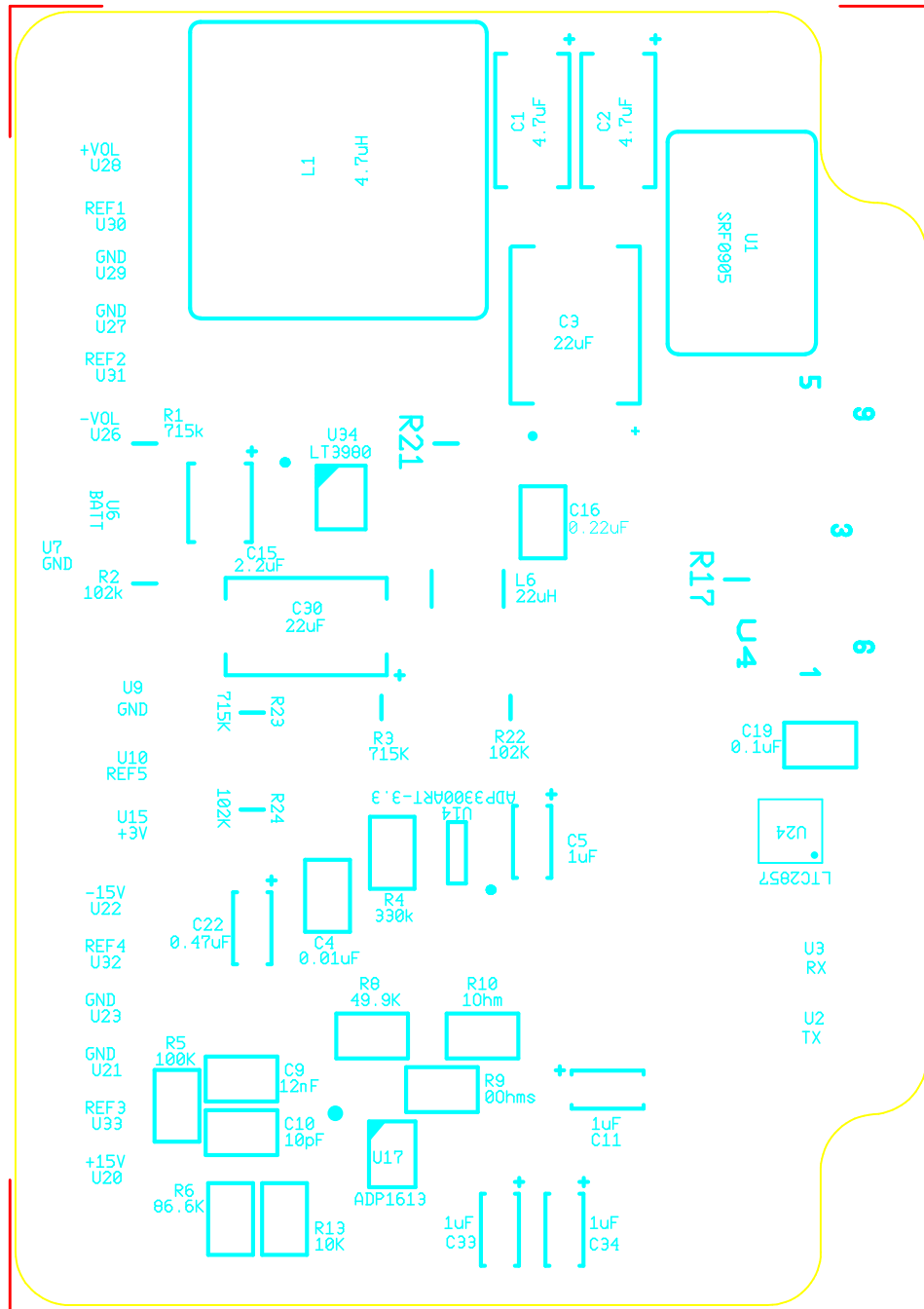
Power Top Copper
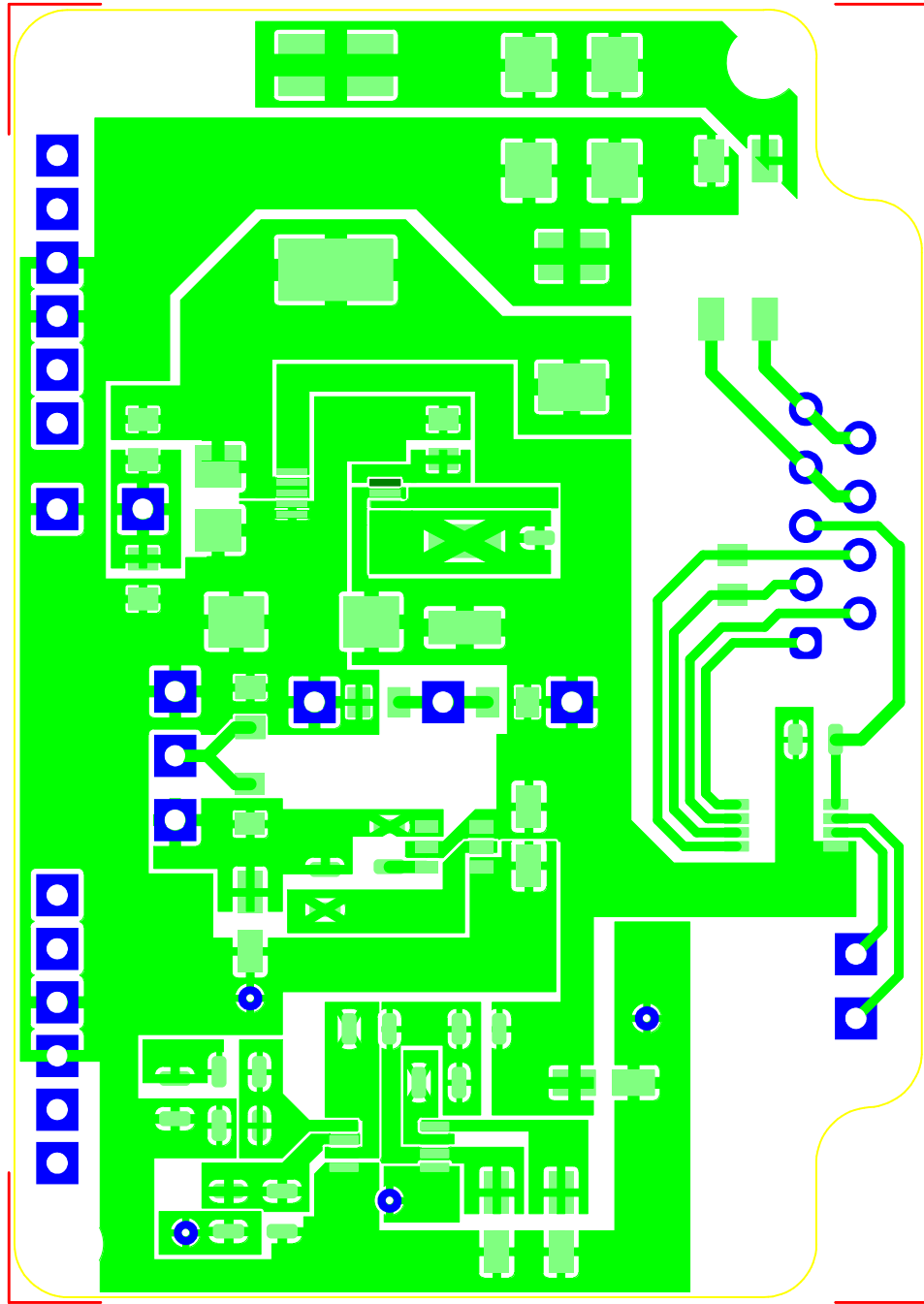
Power Bottom Copper
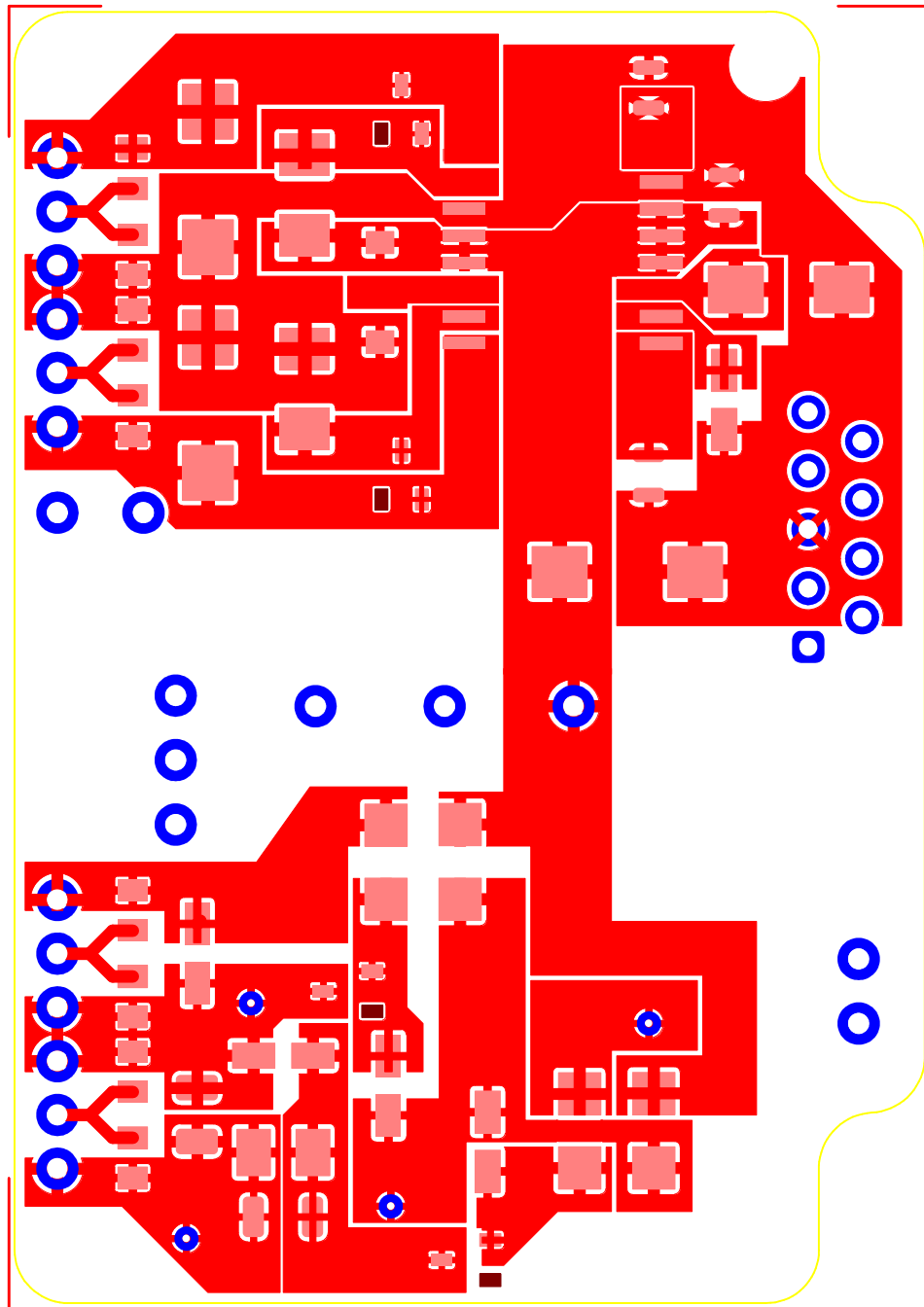
Power Bottome Silkscreen
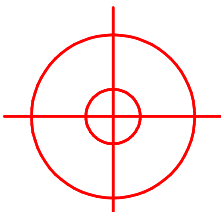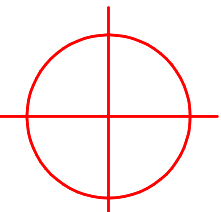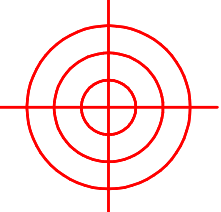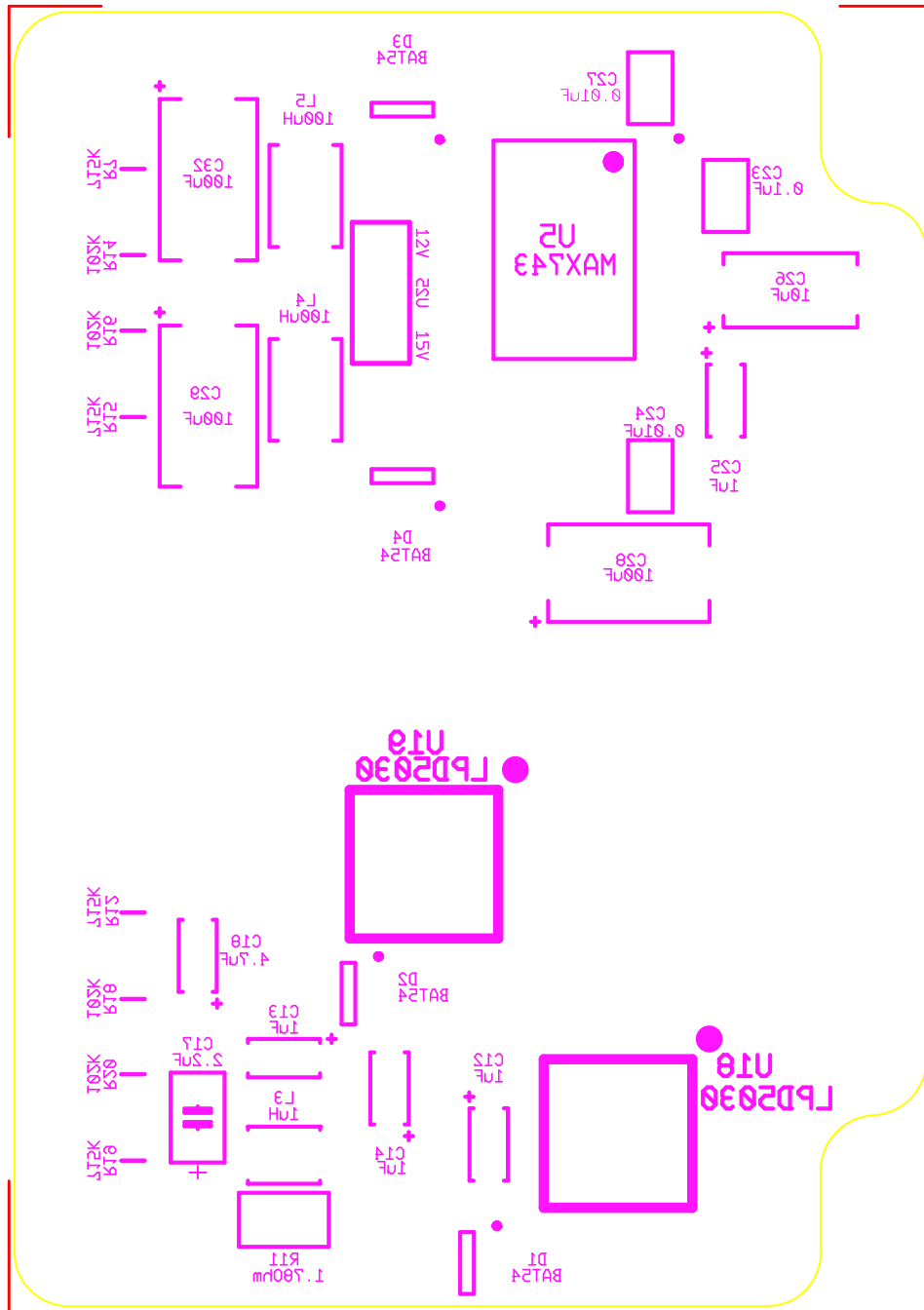
5.3 Microcontroller Code

5.3.1 Sweeping Langmuir Probe Code

Main SLP Code

```c
#include <asf.h>
#include "mtex.h"

#define AD7656_BAUDRATE  12000000UL
//#define ASYNCH_BAUDRATE  115200UL
//#define ASYNCH_BAUDRATE  500000UL
#define ASYNCH_BAUDRATE  3000000UL

#define DAC8581_BAUDRATE 12000000UL

#define SYNCH0          0xCCAA
#define SYNCH1          0xE38C

//#define SWEEPSTEPS     2621
#define SWEEPSTEPS       656
#define CHANNELS       2
//#define OVERSAMPLE     64
#define OVERSAMPLE       1

#define HK_ADC_WORDS   3
#define PAYLOADTAG       SLP_TAG | TAG_01

#include "adamClocks.h"
#include "ad7656.h"
#include "asynch.h"
#include "dac8581.h"
#include "hk.h"
```

```
static const uint8_t tempOrder[] = {4, 5, 3}; //INT1, INT2, EXT

const int16_t sweepVals[656] = {-32750, -32650, -32550, -32450, -32350,
    -32250, -32150, -32050, -31950, -31850, -31750, -31650, -31550, -31450,
    -31350, -31250, -31150, -31050, -30950, -30850, -30750, -30650, -30550,
    -30450, -30350, -30250, -30150, -30050, -29950, -29850, -29750, -29650,
    -29550, -29450, -29350, -29250, -29150, -29050, -28950, -28850, -28750,
    -28650, -28550, -28450, -28350, -28250, -28150, -28050, -27950, -27850,
    -27750, -27650, -27550, -27450, -27350, -27250, -27150, -27050, -26950,
    -26850, -26750, -26650, -26550, -26450, -26350, -26250, -26150, -26050,
    -25950, -25850, -25750, -25650, -25550, -25450, -25350, -25250, -25150,
    -25050, -24950, -24850, -24750, -24650, -24550, -24450, -24350, -24250,
    -24150, -24050, -23950, -23850, -23750, -23650, -23550, -23450, -23350,
    -23250, -23150, -23050, -22950, -22850, -22750, -22650, -22550, -22450,
    -22350, -22250, -22150, -22050, -21950, -21850, -21750, -21650, -21550,
    -21450, -21350, -21250, -21150, -21050, -20950, -20850, -20750, -20650,
    -20550, -20450, -20350, -20250, -20150, -20050, -19950, -19850, -19750,
    -19650, -19550, -19450, -19350, -19250, -19150, -19050, -18950, -18850,
    -18750, -18650, -18550, -18450, -18350, -18250, -18150, -18050, -17950,
    -17850, -17750, -17650, -17550, -17450, -17350, -17250, -17150, -17050,
    -16950, -16850, -16750, -16650, -16550, -16450, -16350, -16250, -16150,
    -16050, -15950, -15850, -15750, -15650, -15550, -15450, -15350, -15250,
```

```
-15150, -15050, -14950, -14850, -14750, -14650, -14550, -14450, -14350,
    -14250, -14150, -14050, -13950, -13850, -13750, -13650, -13550, -13450,
    -13350, -13250, -13150, -13050, -12950, -12850, -12750, -12650, -12550,
    -12450, -12350, -12250, -12150, -12050, -11950, -11850, -11750, -11650,
    -11550, -11450, -11350, -11250, -11150, -11050, -10950, -10850, -10750,
    -10650, -10550, -10450, -10350, -10250, -10150, -10050, -9950, -9850,
    -9750, -9650, -9550, -9450, -9350, -9250, -9150, -9050, -8950, -8850,
    -8750, -8650, -8550, -8450, -8350, -8250, -8150, -8050, -7950, -7850,
    -7750, -7650, -7550, -7450, -7350, -7250, -7150, -7050, -6950, -6850,
    -6750, -6650, -6550, -6450, -6350, -6250, -6150, -6050, -5950, -5850,
    -5750, -5650, -5550, -5450, -5350, -5250, -5150, -5050, -4950, -4850,
    -4750, -4650, -4550, -4450, -4350, -4250, -4150, -4050, -3950, -3850,
    -3750, -3650, -3550, -3450, -3350, -3250, -3150, -3050, -2950, -2850,
    -2750, -2650, -2550, -2450, -2350, -2250, -2150, -2050, -1950, -1850,
    -1750, -1650, -1550, -1450, -1350, -1250, -1150, -1050, -950, -850,
    -750, -650, -550, -450, -350, -250, -150, -50, +50, +150, +250, +350,
    +450, +550, +650, +750, +850, +950, +1050, +1150, +1250, +1350, +1450,
    +1550, +1650, +1750, +1850, +1950, +2050, +2150, +2250, +2350, +2450,
    +2550, +2650, +2750, +2850, +2950, +3050, +3150, +3250, +3350, +3450,
    +3550, +3650, +3750, +3850, +3950, +4050, +4150, +4250, +4350, +4450,
    +4550, +4650, +4750, +4850, +4950, +5050, +5150, +5250, +5350, +5450,
    +5550, +5650, +5750, +5850, +5950, +6050, +6150, +6250, +6350, +6450,
    +6550, +6650, +6750, +6850, +6950, +7050, +7150, +7250, +7350, +7450,
    +7550, +7650, +7750, +7850, +7950, +8050, +8150, +8250, +8350, +8450,
    +8550, +8650, +8750, +8850, +8950, +9050, +9150, +9250, +9350, +9450,
    +9550, +9650, +9750, +9850, +9950, +10050, +10150, +10250, +10350,
    +10450, +10550, +10650, +10750, +10850, +10950, +11050, +11150, +11250,
    +11350, +11450, +11550, +11650, +11750, +11850, +11950, +12050, +12150,
    +12250, +12350, +12450, +12550, +12650, +12750, +12850, +12950, +13050,
    +13150, +13250, +13350, +13450, +13550, +13650, +13750, +13850, +13950,
    +14050, +14150, +14250, +14350, +14450, +14550, +14650, +14750, +14850,
    +14950, +15050, +15150, +15250, +15350, +15450, +15550, +15650, +15750,
    +15850, +15950, +16050, +16150, +16250, +16350, +16450, +16550, +16650,
```

```
+16750, +16850, +16950, +17050, +17150, +17250, +17350, +17450, +17550,
    +17650, +17750, +17850, +17950, +18050, +18150, +18250, +18350, +18450,
    +18550, +18650, +18750, +18850, +18950, +19050, +19150, +19250, +19350,
    +19450, +19550, +19650, +19750, +19850, +19950, +20050, +20150, +20250,
    +20350, +20450, +20550, +20650, +20750, +20850, +20950, +21050, +21150,
    +21250, +21350, +21450, +21550, +21650, +21750, +21850, +21950, +22050,
    +22150, +22250, +22350, +22450, +22550, +22650, +22750, +22850, +22950,
    +23050, +23150, +23250, +23350, +23450, +23550, +23650, +23750, +23850,
    +23950, +24050, +24150, +24250, +24350, +24450, +24550, +24650, +24750,
    +24850, +24950, +25050, +25150, +25250, +25350, +25450, +25550, +25650,
    +25750, +25850, +25950, +26050, +26150, +26250, +26350, +26450, +26550,
    +26650, +26750, +26850, +26950, +27050, +27150, +27250, +27350, +27450,
    +27550, +27650, +27750, +27850, +27950, +28050, +28150, +28250, +28350,
    +28450, +28550, +28650, +28750, +28850, +28950, +29050, +29150, +29250,
    +29350, +29450, +29550, +29650, +29750, +29850, +29950, +30050, +30150,
    +30250, +30350, +30450, +30550, +30650, +30750, +30850, +30950, +31050,
    +31150, +31250, +31350, +31450, +31550, +31650, +31750, +31850, +31950,
    +32050, +32150, +32250, +32350, +32450, +32550, +32650,+32750};

int main (void)
{
    uint16_t step;
    uint8_t i;
    int16_t data[CHANNELS];
    int16_t hkData[HK_ADC_CONVERSIONS];

    board_init();
    adamClocks_setup();
    ad7656_setup();
    asynch_setup();
    dac8581_setup();
    hk_setup();
```

```c
while (1)
{

   //Send Frame synch
   asynch_word(SYNCH0);
   asynch_word(SYNCH1);

   //Do upward sweep
   for (step=0 ; step<SWEEPSTEPS ; step++)
   {
      dac8581_set(sweepVals[step]);        //Set
      ad7656_convert(data , CHANNELS , OVERSAMPLE);  //Get
      for (i=0 ; i<CHANNELS ; i++) asynch_word(data[i]);  //Give
   }

   //Collect Housekeeping
   hk_get_data(hkData);               //Get
   for (i=0 ; i<HK_ADC_WORDS ; i++) asynch_word(hkData[tempOrder[i]]);
        //Give

   //Do downward sweep
   for (step = 0 ; step<SWEEPSTEPS ; step++)
   {
      dac8581_set(sweepVals[SWEEPSTEPS-step-1]);  //Set
      ad7656_convert(data , CHANNELS , OVERSAMPLE);  //Get
      for (i=0 ; i<CHANNELS ; i++) asynch_word(data[i]);  //Give
   }

   //Collect Housekeeping
   hk_get_data(hkData);               //Get
   for (i=0 ; i<HK_ADC_WORDS ; i++) asynch_word(hkData[tempOrder[i]]);
        //Give
```

```
        //asynch_word(PAYLOADTAG);
    }
}
```

DAC Calibration Code

```
#include <asf.h>
#include "mtex.h"


#define AD7656_BAUDRATE  12000000UL
#define ASYNCH_BAUDRATE  115200UL
#define DAC8581_BAUDRATE 12000000UL


#define SYNCH0         0xCCAA
#define SYNCH1         0xE38C


#define SWEEPSTEPS       716
#define CHANNELS      2
#define OVERSAMPLE       32


#define HK_ADC_WORDS  3
#define PAYLOADTAG       SLP_TAG | TAG_01


#define CALIB_USART           AVR32_USART0


static const uint8_t tempOrder[] = {4, 5, 3}; //INT1, INT2, EXT

const int16_t sweepVals[2621] = { -32750, -32725, -32700, -32675, -32650,
    -32625, -32600, -32575, -32550, -32525, -32500, -32475, -32450, -32425,
    -32400, -32375, -32350, -32325, -32300, -32275, -32250, -32225, -32200,
    -32175, -32150, -32125, -32100, -32075, -32050, -32025, -32000, -31975,
    -31950, -31925, -31900, -31875, -31850, -31825, -31800, -31775, -31750,
    -31725, -31700, -31675, -31650, -31625, -31600, -31575, -31550, -31525,
    -31500, -31475, -31450, -31425, -31400, -31375, -31350, -31325, -31300,
    -31275, -31250, -31225, -31200, -31175, -31150, -31125, -31100, -31075,
    -31050, -31025, -31000, -30975, -30950, -30925, -30900, -30875, -30850,
    -30825, -30800, -30775, -30750, -30725, -30700, -30675, -30650, -30625,
```

-30600, -30575, -30550, -30525, -30500, -30475, -30450, -30425, -30400,
    -30375, -30350, -30325, -30300, -30275, -30250, -30225, -30200, -30175,
    -30150, -30125, -30100, -30075, -30050, -30025, -30000, -29975, -29950,
    -29925, -29900, -29875, -29850, -29825, -29800, -29775, -29750, -29725,
    -29700, -29675, -29650, -29625, -29600, -29575, -29550, -29525, -29500,
    -29475, -29450, -29425, -29400, -29375, -29350, -29325, -29300, -29275,
    -29250, -29225, -29200, -29175, -29150, -29125, -29100, -29075, -29050,
    -29025, -29000, -28975, -28950, -28925, -28900, -28875, -28850, -28825,
    -28800, -28775, -28750, -28725, -28700, -28675, -28650, -28625, -28600,
    -28575, -28550, -28525, -28500, -28475, -28450, -28425, -28400, -28375,
    -28350, -28325, -28300, -28275, -28250, -28225, -28200, -28175, -28150,
    -28125, -28100, -28075, -28050, -28025, -28000, -27975, -27950, -27925,
    -27900, -27875, -27850, -27825, -27800, -27775, -27750, -27725, -27700,
    -27675, -27650, -27625, -27600, -27575, -27550, -27525, -27500, -27475,
    -27450, -27425, -27400, -27375, -27350, -27325, -27300, -27275, -27250,
    -27225, -27200, -27175, -27150, -27125, -27100, -27075, -27050, -27025,
    -27000, -26975, -26950, -26925, -26900, -26875, -26850, -26825, -26800,
    -26775, -26750, -26725, -26700, -26675, -26650, -26625, -26600, -26575,
    -26550, -26525, -26500, -26475, -26450, -26425, -26400, -26375, -26350,
    -26325, -26300, -26275, -26250, -26225, -26200, -26175, -26150, -26125,
    -26100, -26075, -26050, -26025, -26000, -25975, -25950, -25925, -25900,
    -25875, -25850, -25825, -25800, -25775, -25750, -25725, -25700, -25675,
    -25650, -25625, -25600, -25575, -25550, -25525, -25500, -25475, -25450,
    -25425, -25400, -25375, -25350, -25325, -25300, -25275, -25250, -25225,
    -25200, -25175, -25150, -25125, -25100, -25075, -25050, -25025, -25000,
    -24975, -24950, -24925, -24900, -24875, -24850, -24825, -24800, -24775,
    -24750, -24725, -24700, -24675, -24650, -24625, -24600, -24575, -24550,
    -24525, -24500, -24475, -24450, -24425, -24400, -24375, -24350, -24325,
    -24300, -24275, -24250, -24225, -24200, -24175, -24150, -24125, -24100,
    -24075, -24050, -24025, -24000, -23975, -23950, -23925, -23900, -23875,
    -23850, -23825, -23800, -23775, -23750, -23725, -23700, -23675, -23650,
    -23625, -23600, -23575, -23550, -23525, -23500, -23475, -23450, -23425,

-23400, -23375, -23350, -23325, -23300, -23275, -23250, -23225, -23200,
     -23175, -23150, -23125, -23100, -23075, -23050, -23025, -23000, -22975,
     -22950, -22925, -22900, -22875, -22850, -22825, -22800, -22775, -22750,
     -22725, -22700, -22675, -22650, -22625, -22600, -22575, -22550, -22525,
     -22500, -22475, -22450, -22425, -22400, -22375, -22350, -22325, -22300,
     -22275, -22250, -22225, -22200, -22175, -22150, -22125, -22100, -22075,
     -22050, -22025, -22000, -21975, -21950, -21925, -21900, -21875, -21850,
     -21825, -21800, -21775, -21750, -21725, -21700, -21675, -21650, -21625,
     -21600, -21575, -21550, -21525, -21500, -21475, -21450, -21425, -21400,
     -21375, -21350, -21325, -21300, -21275, -21250, -21225, -21200, -21175,
     -21150, -21125, -21100, -21075, -21050, -21025, -21000, -20975, -20950,
     -20925, -20900, -20875, -20850, -20825, -20800, -20775, -20750, -20725,
     -20700, -20675, -20650, -20625, -20600, -20575, -20550, -20525, -20500,
     -20475, -20450, -20425, -20400, -20375, -20350, -20325, -20300, -20275,
     -20250, -20225, -20200, -20175, -20150, -20125, -20100, -20075, -20050,
     -20025, -20000, -19975, -19950, -19925, -19900, -19875, -19850, -19825,
     -19800, -19775, -19750, -19725, -19700, -19675, -19650, -19625, -19600,
     -19575, -19550, -19525, -19500, -19475, -19450, -19425, -19400, -19375,
     -19350, -19325, -19300, -19275, -19250, -19225, -19200, -19175, -19150,
     -19125, -19100, -19075, -19050, -19025, -19000, -18975, -18950, -18925,
     -18900, -18875, -18850, -18825, -18800, -18775, -18750, -18725, -18700,
     -18675, -18650, -18625, -18600, -18575, -18550, -18525, -18500, -18475,
     -18450, -18425, -18400, -18375, -18350, -18325, -18300, -18275, -18250,
     -18225, -18200, -18175, -18150, -18125, -18100, -18075, -18050, -18025,
     -18000, -17975, -17950, -17925, -17900, -17875, -17850, -17825, -17800,
     -17775, -17750, -17725, -17700, -17675, -17650, -17625, -17600, -17575,
     -17550, -17525, -17500, -17475, -17450, -17425, -17400, -17375, -17350,
     -17325, -17300, -17275, -17250, -17225, -17200, -17175, -17150, -17125,
     -17100, -17075, -17050, -17025, -17000, -16975, -16950, -16925, -16900,
     -16875, -16850, -16825, -16800, -16775, -16750, -16725, -16700, -16675,
     -16650, -16625, -16600, -16575, -16550, -16525, -16500, -16475, -16450,
     -16425, -16400, -16375, -16350, -16325, -16300, -16275, -16250, -16225,
     -16200, -16175, -16150, -16125, -16100, -16075, -16050, -16025, -16000,

-15975, -15950, -15925, -15900, -15875, -15850, -15825, -15800, -15775,
    -15750, -15725, -15700, -15675, -15650, -15625, -15600, -15575, -15550,
    -15525, -15500, -15475, -15450, -15425, -15400, -15375, -15350, -15325,
    -15300, -15275, -15250, -15225, -15200, -15175, -15150, -15125, -15100,
    -15075, -15050, -15025, -15000, -14975, -14950, -14925, -14900, -14875,
    -14850, -14825, -14800, -14775, -14750, -14725, -14700, -14675, -14650,
    -14625, -14600, -14575, -14550, -14525, -14500, -14475, -14450, -14425,
    -14400, -14375, -14350, -14325, -14300, -14275, -14250, -14225, -14200,
    -14175, -14150, -14125, -14100, -14075, -14050, -14025, -14000, -13975,
    -13950, -13925, -13900, -13875, -13850, -13825, -13800, -13775, -13750,
    -13725, -13700, -13675, -13650, -13625, -13600, -13575, -13550, -13525,
    -13500, -13475, -13450, -13425, -13400, -13375, -13350, -13325, -13300,
    -13275, -13250, -13225, -13200, -13175, -13150, -13125, -13100, -13075,
    -13050, -13025, -13000, -12975, -12950, -12925, -12900, -12875, -12850,
    -12825, -12800, -12775, -12750, -12725, -12700, -12675, -12650, -12625,
    -12600, -12575, -12550, -12525, -12500, -12475, -12450, -12425, -12400,
    -12375, -12350, -12325, -12300, -12275, -12250, -12225, -12200, -12175,
    -12150, -12125, -12100, -12075, -12050, -12025, -12000, -11975, -11950,
    -11925, -11900, -11875, -11850, -11825, -11800, -11775, -11750, -11725,
    -11700, -11675, -11650, -11625, -11600, -11575, -11550, -11525, -11500,
    -11475, -11450, -11425, -11400, -11375, -11350, -11325, -11300, -11275,
    -11250, -11225, -11200, -11175, -11150, -11125, -11100, -11075, -11050,
    -11025, -11000, -10975, -10950, -10925, -10900, -10875, -10850, -10825,
    -10800, -10775, -10750, -10725, -10700, -10675, -10650, -10625, -10600,
    -10575, -10550, -10525, -10500, -10475, -10450, -10425, -10400, -10375,
    -10350, -10325, -10300, -10275, -10250, -10225, -10200, -10175, -10150,
    -10125, -10100, -10075, -10050, -10025, -10000, -9975, -9950, -9925,
    -9900, -9875, -9850, -9825, -9800, -9775, -9750, -9725, -9700, -9675,
    -9650, -9625, -9600, -9575, -9550, -9525, -9500, -9475, -9450, -9425,
    -9400, -9375, -9350, -9325, -9300, -9275, -9250, -9225, -9200, -9175,
    -9150, -9125, -9100, -9075, -9050, -9025, -9000, -8975, -8950, -8925,
    -8900, -8875, -8850, -8825, -8800, -8775, -8750, -8725, -8700, -8675,
    -8650, -8625, -8600, -8575, -8550, -8525, -8500, -8475, -8450, -8425,

```
-8400, -8375, -8350, -8325, -8300, -8275, -8250, -8225, -8200, -8175,
    -8150, -8125, -8100, -8075, -8050, -8025, -8000, -7975, -7950, -7925,
    -7900, -7875, -7850, -7825, -7800, -7775, -7750, -7725, -7700, -7675,
    -7650, -7625, -7600, -7575, -7550, -7525, -7500, -7475, -7450, -7425,
    -7400, -7375, -7350, -7325, -7300, -7275, -7250, -7225, -7200, -7175,
    -7150, -7125, -7100, -7075, -7050, -7025, -7000, -6975, -6950, -6925,
    -6900, -6875, -6850, -6825, -6800, -6775, -6750, -6725, -6700, -6675,
    -6650, -6625, -6600, -6575, -6550, -6525, -6500, -6475, -6450, -6425,
    -6400, -6375, -6350, -6325, -6300, -6275, -6250, -6225, -6200, -6175,
    -6150, -6125, -6100, -6075, -6050, -6025, -6000, -5975, -5950, -5925,
    -5900, -5875, -5850, -5825, -5800, -5775, -5750, -5725, -5700, -5675,
    -5650, -5625, -5600, -5575, -5550, -5525, -5500, -5475, -5450, -5425,
    -5400, -5375, -5350, -5325, -5300, -5275, -5250, -5225, -5200, -5175,
    -5150, -5125, -5100, -5075, -5050, -5025, -5000, -4975, -4950, -4925,
    -4900, -4875, -4850, -4825, -4800, -4775, -4750, -4725, -4700, -4675,
    -4650, -4625, -4600, -4575, -4550, -4525, -4500, -4475, -4450, -4425,
    -4400, -4375, -4350, -4325, -4300, -4275, -4250, -4225, -4200, -4175,
    -4150, -4125, -4100, -4075, -4050, -4025, -4000, -3975, -3950, -3925,
    -3900, -3875, -3850, -3825, -3800, -3775, -3750, -3725, -3700, -3675,
    -3650, -3625, -3600, -3575, -3550, -3525, -3500, -3475, -3450, -3425,
    -3400, -3375, -3350, -3325, -3300, -3275, -3250, -3225, -3200, -3175,
    -3150, -3125, -3100, -3075, -3050, -3025, -3000, -2975, -2950, -2925,
    -2900, -2875, -2850, -2825, -2800, -2775, -2750, -2725, -2700, -2675,
    -2650, -2625, -2600, -2575, -2550, -2525, -2500, -2475, -2450, -2425,
    -2400, -2375, -2350, -2325, -2300, -2275, -2250, -2225, -2200, -2175,
    -2150, -2125, -2100, -2075, -2050, -2025, -2000, -1975, -1950, -1925,
    -1900, -1875, -1850, -1825, -1800, -1775, -1750, -1725, -1700, -1675,
    -1650, -1625, -1600, -1575, -1550, -1525, -1500, -1475, -1450, -1425,
    -1400, -1375, -1350, -1325, -1300, -1275, -1250, -1225, -1200, -1175,
    -1150, -1125, -1100, -1075, -1050, -1025, -1000,  -975,  -950,  -925,
     -900,  -875,  -850,  -825,  -800,  -775,  -750,  -725,  -700,  -675,
     -650,  -625,  -600,  -575,  -550,  -525,  -500,  -475,  -450,  -425,
     -400,  -375,  -350,  -325,  -300,  -275,  -250,  -225,  -200,  -175,
```

```
-150,  -125,  -100,   -75,   -50,   -25,      +0,    +25,    +50,    +75,
    +100,   +125,   +150,   +175,   +200,   +225,   +250,   +275,   +300,   +325,
    +350,   +375,   +400,   +425,   +450,   +475,   +500,   +525,   +550,   +575,
    +600,   +625,   +650,   +675,   +700,   +725,   +750,   +775,   +800,   +825,
    +850,   +875,   +900,   +925,   +950,   +975,  +1000,  +1025,  +1050,  +1075,
   +1100,  +1125,  +1150,  +1175,  +1200,  +1225,  +1250,  +1275,  +1300,  +1325,
   +1350,  +1375,  +1400,  +1425,  +1450,  +1475,  +1500,  +1525,  +1550,  +1575,
   +1600,  +1625,  +1650,  +1675,  +1700,  +1725,      +1750,  +1775,  +1800,
   +1825,  +1850,  +1875,  +1900,  +1925,  +1950,  +1975,  +2000,  +2025,  +2050,
   +2075,  +2100,  +2125,  +2150,  +2175,  +2200,  +2225,  +2250,  +2275,  +2300,
   +2325,  +2350,  +2375,  +2400,  +2425,  +2450,  +2475,  +2500,  +2525,  +2550,
   +2575,  +2600,  +2625,  +2650,  +2675,  +2700,  +2725,  +2750,  +2775,  +2800,
   +2825,  +2850,  +2875,  +2900,  +2925,  +2950,  +2975,  +3000,  +3025,  +3050,
   +3075,  +3100,  +3125,  +3150,  +3175,  +3200,  +3225,  +3250,  +3275,  +3300,
   +3325,  +3350,  +3375,  +3400,  +3425,  +3450,  +3475,  +3500,  +3525,  +3550,
   +3575,  +3600,  +3625,  +3650,  +3675,  +3700,  +3725,  +3750,  +3775,  +3800,
   +3825,  +3850,  +3875,  +3900,  +3925,  +3950,  +3975,  +4000,  +4025,  +4050,
   +4075,  +4100,  +4125,  +4150,  +4175,  +4200,  +4225,  +4250,  +4275,  +4300,
   +4325,  +4350,  +4375,  +4400,  +4425,  +4450,  +4475,  +4500,  +4525,  +4550,
   +4575,  +4600,  +4625,  +4650,  +4675,  +4700,  +4725,  +4750,  +4775,  +4800,
   +4825,  +4850,  +4875,  +4900,  +4925,  +4950,  +4975,  +5000,  +5025,  +5050,
   +5075,  +5100,  +5125,  +5150,  +5175,  +5200,  +5225,  +5250,  +5275,  +5300,
   +5325,  +5350,  +5375,  +5400,  +5425,  +5450,  +5475,  +5500,  +5525,  +5550,
   +5575,  +5600,  +5625,  +5650,  +5675,  +5700,  +5725,  +5750,  +5775,  +5800,
   +5825,  +5850,  +5875,  +5900,  +5925,  +5950,  +5975,  +6000,  +6025,  +6050,
   +6075,  +6100,  +6125,  +6150,  +6175,  +6200,  +6225,  +6250,  +6275,  +6300,
   +6325,  +6350,  +6375,  +6400,  +6425,  +6450,  +6475,  +6500,  +6525,  +6550,
   +6575,  +6600,  +6625,  +6650,  +6675,  +6700,  +6725,  +6750,  +6775,  +6800,
   +6825,  +6850,  +6875,  +6900,  +6925,  +6950,  +6975,  +7000,  +7025,  +7050,
   +7075,  +7100,  +7125,  +7150,  +7175,  +7200,  +7225,  +7250,  +7275,  +7300,
   +7325,  +7350,  +7375,  +7400,  +7425,  +7450,  +7475,  +7500,  +7525,  +7550,
   +7575,  +7600,  +7625,  +7650,  +7675,  +7700,  +7725,  +7750,  +7775,  +7800,
   +7825,  +7850,  +7875,  +7900,  +7925,  +7950,  +7975,  +8000,  +8025,  +8050,
```

+8075, +8100, +8125, +8150, +8175, +8200, +8225, +8250, +8275, +8300,
+8325, +8350, +8375, +8400, +8425, +8450, +8475, +8500, +8525, +8550,
+8575, +8600, +8625, +8650, +8675, +8700, +8725, +8750, +8775, +8800,
+8825, +8850, +8875, +8900, +8925, +8950, +8975, +9000, +9025, +9050,
+9075, +9100, +9125, +9150, +9175, +9200, +9225, +9250, +9275, +9300,
+9325, +9350, +9375, +9400, +9425, +9450, +9475, +9500, +9525, +9550,
+9575, +9600, +9625, +9650, +9675, +9700, +9725, +9750, +9775, +9800,
+9825, +9850, +9875, +9900, +9925, +9950, +9975, +10000, +10025,
+10050, +10075, +10100, +10125, +10150, +10175, +10200, +10225, +10250,
+10275, +10300, +10325, +10350, +10375, +10400, +10425, +10450, +10475,
+10500, +10525, +10550, +10575, +10600, +10625, +10650, +10675, +10700,
+10725, +10750, +10775, +10800, +10825, +10850, +10875, +10900, +10925,
+10950, +10975, +11000, +11025, +11050, +11075, +11100, +11125, +11150,
+11175, +11200, +11225, +11250, +11275, +11300, +11325, +11350, +11375,
+11400, +11425, +11450, +11475, +11500, +11525, +11550, +11575, +11600,
+11625, +11650, +11675, +11700, +11725, +11750, +11775, +11800, +11825,
+11850, +11875, +11900, +11925, +11950, +11975, +12000, +12025, +12050,
+12075, +12100, +12125, +12150, +12175, +12200, +12225, +12250, +12275,
+12300, +12325, +12350, +12375, +12400, +12425, +12450, +12475, +12500,
+12525, +12550, +12575, +12600, +12625, +12650, +12675, +12700, +12725,
+12750, +12775, +12800, +12825, +12850, +12875, +12900, +12925, +12950,
+12975, +13000, +13025, +13050, +13075, +13100, +13125, +13150, +13175,
+13200, +13225, +13250, +13275, +13300, +13325, +13350, +13375, +13400,
+13425, +13450, +13475, +13500, +13525, +13550, +13575, +13600, +13625,
+13650, +13675, +13700, +13725, +13750, +13775, +13800, +13825, +13850,
+13875, +13900, +13925, +13950, +13975, +14000, +14025, +14050, +14075,
+14100, +14125, +14150, +14175, +14200, +14225, +14250, +14275, +14300,
+14325, +14350, +14375, +14400, +14425, +14450, +14475, +14500, +14525,
+14550, +14575, +14600, +14625, +14650, +14675, +14700, +14725, +14750,
+14775, +14800, +14825, +14850, +14875, +14900, +14925, +14950, +14975,
+15000, +15025, +15050, +15075, +15100, +15125, +15150, +15175, +15200,
+15225, +15250, +15275, +15300, +15325, +15350, +15375, +15400, +15425,
+15450, +15475, +15500, +15525, +15550, +15575, +15600, +15625, +15650,

+15675, +15700, +15725, +15750, +15775, +15800, +15825, +15850, +15875,
+15900, +15925, +15950, +15975, +16000, +16025, +16050, +16075, +16100,
+16125, +16150, +16175, +16200, +16225, +16250, +16275, +16300, +16325,
+16350, +16375, +16400, +16425, +16450, +16475, +16500, +16525, +16550,
+16575, +16600, +16625, +16650, +16675, +16700, +16725, +16750, +16775,
+16800, +16825, +16850, +16875, +16900, +16925, +16950, +16975, +17000,
+17025, +17050, +17075, +17100, +17125, +17150, +17175, +17200, +17225,
+17250, +17275, +17300, +17325, +17350, +17375, +17400, +17425, +17450,
+17475, +17500, +17525, +17550, +17575, +17600, +17625, +17650, +17675,
+17700, +17725, +17750, +17775, +17800, +17825, +17850, +17875, +17900,
+17925, +17950, +17975, +18000, +18025, +18050, +18075, +18100, +18125,
+18150, +18175, +18200, +18225, +18250, +18275, +18300, +18325, +18350,
+18375, +18400, +18425, +18450, +18475, +18500, +18525, +18550, +18575,
+18600, +18625, +18650, +18675, +18700, +18725, +18750, +18775, +18800,
+18825, +18850, +18875, +18900, +18925, +18950, +18975, +19000, +19025,
+19050, +19075, +19100, +19125, +19150, +19175, +19200, +19225, +19250,
+19275, +19300, +19325, +19350, +19375, +19400, +19425, +19450, +19475,
+19500, +19525, +19550, +19575, +19600, +19625, +19650, +19675, +19700,
+19725, +19750, +19775, +19800, +19825, +19850, +19875, +19900, +19925,
+19950, +19975, +20000, +20025, +20050, +20075, +20100, +20125, +20150,
+20175, +20200, +20225, +20250, +20275, +20300, +20325, +20350, +20375,
+20400, +20425, +20450, +20475, +20500, +20525, +20550, +20575, +20600,
+20625, +20650, +20675, +20700, +20725, +20750, +20775, +20800, +20825,
+20850, +20875, +20900, +20925, +20950, +20975, +21000, +21025, +21050,
+21075, +21100, +21125, +21150, +21175, +21200, +21225, +21250, +21275,
+21300, +21325, +21350, +21375, +21400, +21425, +21450, +21475, +21500,
+21525, +21550, +21575, +21600, +21625, +21650, +21675, +21700, +21725,
+21750, +21775, +21800, +21825, +21850, +21875, +21900, +21925, +21950,
+21975, +22000, +22025, +22050, +22075, +22100, +22125, +22150, +22175,
+22200, +22225, +22250, +22275, +22300, +22325, +22350, +22375, +22400,
+22425, +22450, +22475, +22500, +22525, +22550, +22575, +22600, +22625,
+22650, +22675, +22700, +22725, +22750, +22775, +22800, +22825, +22850,

+22875, +22900, +22925, +22950, +22975, +23000, +23025, +23050, +23075,
    +23100, +23125, +23150, +23175, +23200, +23225, +23250, +23275, +23300,
    +23325, +23350, +23375, +23400, +23425, +23450, +23475, +23500, +23525,
    +23550, +23575, +23600, +23625, +23650, +23675, +23700, +23725, +23750,
    +23775, +23800, +23825, +23850, +23875, +23900, +23925, +23950, +23975,
    +24000, +24025, +24050, +24075, +24100, +24125, +24150, +24175, +24200,
    +24225, +24250, +24275, +24300, +24325, +24350, +24375, +24400, +24425,
    +24450, +24475, +24500, +24525, +24550, +24575, +24600, +24625, +24650,
    +24675, +24700, +24725, +24750, +24775, +24800, +24825, +24850, +24875,
    +24900, +24925, +24950, +24975, +25000, +25025, +25050, +25075, +25100,
    +25125, +25150, +25175, +25200, +25225, +25250, +25275, +25300, +25325,
    +25350, +25375, +25400, +25425, +25450, +25475, +25500, +25525, +25550,
    +25575, +25600, +25625, +25650, +25675, +25700, +25725, +25750, +25775,
    +25800, +25825, +25850, +25875, +25900, +25925, +25950, +25975, +26000,
    +26025, +26050, +26075, +26100, +26125, +26150, +26175, +26200, +26225,
    +26250, +26275, +26300, +26325, +26350, +26375, +26400, +26425, +26450,
    +26475, +26500, +26525, +26550, +26575, +26600, +26625, +26650, +26675,
    +26700, +26725, +26750, +26775, +26800, +26825, +26850, +26875, +26900,
    +26925, +26950, +26975, +27000, +27025, +27050, +27075, +27100, +27125,
    +27150, +27175, +27200, +27225, +27250, +27275, +27300, +27325, +27350,
    +27375, +27400, +27425, +27450, +27475, +27500, +27525, +27550, +27575,
    +27600, +27625, +27650, +27675, +27700, +27725, +27750, +27775, +27800,
    +27825, +27850, +27875, +27900, +27925, +27950, +27975, +28000, +28025,
    +28050, +28075, +28100, +28125, +28150, +28175, +28200, +28225, +28250,
    +28275, +28300, +28325, +28350, +28375, +28400, +28425, +28450, +28475,
    +28500, +28525, +28550, +28575, +28600, +28625, +28650, +28675, +28700,
    +28725, +28750, +28775, +28800, +28825, +28850, +28875, +28900, +28925,
    +28950, +28975, +29000, +29025, +29050, +29075, +29100, +29125, +29150,
    +29175, +29200, +29225, +29250, +29275, +29300, +29325, +29350, +29375,
    +29400, +29425, +29450, +29475, +29500, +29525, +29550, +29575, +29600,
    +29625, +29650, +29675, +29700, +29725, +29750, +29775, +29800, +29825,
    +29850, +29875, +29900, +29925, +29950, +29975, +30000, +30025, +30050,

```
+30075, +30100, +30125, +30150, +30175, +30200, +30225, +30250, +30275,
    +30300, +30325, +30350, +30375, +30400, +30425, +30450, +30475, +30500,
    +30525, +30550, +30575, +30600, +30625, +30650, +30675, +30700, +30725,
    +30750, +30775, +30800, +30825, +30850, +30875, +30900, +30925, +30950,
    +30975, +31000, +31025, +31050, +31075, +31100, +31125, +31150, +31175,
    +31200, +31225, +31250, +31275, +31300, +31325, +31350, +31375, +31400,
    +31425, +31450, +31475, +31500, +31525, +31550, +31575, +31600, +31625,
    +31650, +31675, +31700, +31725, +31750, +31775, +31800, +31825, +31850,
    +31875, +31900, +31925, +31950, +31975, +32000, +32025, +32050, +32075,
    +32100, +32125, +32150, +32175, +32200, +32225, +32250, +32275, +32300,
    +32325, +32350, +32375, +32400, +32425, +32450, +32475, +32500, +32525,
    +32550, +32575, +32600, +32625, +32650, +32675, +32700, +32725,+32750};

__attribute__((__interrupt__)) void rx_interrupt ();

#include "adamClocks.h"
#include "ad7656.h"
#include "asynch.h"
#include "dac8581.h"
#include "hk.h"

volatile int c;
volatile int16_t step = 0;
volatile int16_t hkData[];

__attribute__((__interrupt__)) void rx_interrupt ()
{
   usart_read_char(&CALIB_USART, &c);
   if (c == 'd')
   {
      step++;
      if (step == 716)
      {
```

```c
        step = 0;
      }
  }
  else if (c == 'a')
  {
    step--;
    if (step == -1)
    {
      step = 716;
    }
  }
  else if (c == 'w')
  {
    step = 715;
  }
  else if (c == 's')
  {
    step = 0;
  }

  for (int i=0 ; i<6 ; i++) asynch_word(hkData[i]); //Give
};


int main (void)
{
  //uint16_t step;
  //uint8_t i;
  int16_t data[CHANNELS];
  hkData[HK_ADC_CONVERSIONS];

  board_init();
  adamClocks_setup();
```

```c
ad7656_setup();
asynch_setup();
dac8581_setup();
hk_setup();

while (1)
{

   //Send Frame synch
   //asynch_word(SYNCH0);
   //asynch_word(SYNCH1);

   //Do upward sweep
   //for (step=0 ; step<SWEEPSTEPS ; step++)
   {
      dac8581_set(sweepVals[step]);      //Set
      ad7656_convert(data , CHANNELS , OVERSAMPLE);  //Get
      //for (i=0 ; i<CHANNELS ; i++) asynch_word(data[i]);  //Give
   }

   //Collect Housekeeping
   hk_get_data(hkData);              //Get

   //asynch_word(PAYLOADTAG);
}
}
```

ADC Code

```c
/* ad7656.h  by   Zachary Laurencio
   Last modified 2014-06-04
   Functions to use the AD7656 16-bit ADC.
   Designed to communicate using USART1 in MSPI mode


   ********************************QUICKSTART**********************************


   REQUIRES:     gpio.h
                 usart.h


   GLOBAL VARS:    int16_t data[] (where data is the destination array
      indexed for however many channels)


   SETUP:        Call ad7656_setup() to configure USART1


   USE:        Call ad7656_convert(int16_t data[],int8_t channels,uint8_t
      oversample)
              Where data will hold returned values, channels tells how many
                 channels to receive
              and oversample is how many averages to take


   **************************************************************************
*/

#ifndef AD7656_H
#define AD7656_H

#include <avr32/io.h>
#include "board.h"
#include "gpio.h"
#include "usart.h"
```

```
#define ADC_USART          AVR32_USART1
#define ADC_RXD_PIN          AVR32_USART1_RXD_1_PIN //pin 54
#define ADC_RXD_FUNCTION     AVR32_USART1_RXD_1_FUNCTION
#define ADC_CLK_PIN          AVR32_USART1_CLK_1_PIN //pin 55
#define ADC_CLK_FUNCTION     AVR32_USART1_CLK_1_FUNCTION
#define ADC_CS_PIN           AVR32_USART1_RTS_1_PIN //pin 56
#define ADC_CS_FUNCTION      GPIO_DIR_OUTPUT
#define ADC_BUSY_PIN        AVR32_USART1_TXD_1_PIN //pin53
#define ADC_BUSY_FUNCTION    GPIO_DIR_INPUT
#define ADC_RESET_PIN         AVR32_PIN_PC20 //pin 44
#define ADC_RESET_FUNCTION   GPIO_DIR_OUTPUT

//Prototypes
void ad7656_setup(void);
void ad7656_convert(volatile int16_t data[] , int8_t channels , uint8_t
   oversample);

//Code per se
void ad7656_setup(void)
{

   static const usart_spi_options_t myUsart =
   {
      .baudrate = AD7656_BAUDRATE,
      .charlength = 8,
      .spimode = 2  //NOTE! OLD VERSIONS MAY ERRONIOUSLY USE SPI MODE 0!!!!
   };

   usart_init_spi_master(&(ADC_USART),&myUsart,48000000);

   gpio_enable_module_pin(ADC_RXD_PIN,ADC_RXD_FUNCTION); //configures pins
      for non-GPIO use
```

```
    gpio_enable_module_pin(ADC_CLK_PIN,ADC_CLK_FUNCTION);


    gpio_configure_pin(ADC_CS_PIN,ADC_CS_FUNCTION);   //Configures pin for
        GPIO use as output
    gpio_set_pin_high(ADC_CS_PIN);


    gpio_enable_gpio_pin (ADC_RESET_PIN);          //configure reset pin for
        output
    gpio_set_gpio_pin(ADC_RESET_PIN);              //reset ADC
    gpio_clr_gpio_pin(ADC_RESET_PIN);              //Reactivate ADC

}


//Reads data from ADC. Collectes channels*16 bits of data. Averages
    'oversample' individual measuremnets per channel
//data[] must be large enough to hold returned data, this is your
    responsibility
void ad7656_convert(volatile int16_t data[] , int8_t channels , uint8_t
    oversample)
{
    int8_t rHigh, rLow;
    int8_t channel;
    uint8_t sample;
    int32_t accumulator[channels];

    for (channel=0 ; channel<channels ; channel++) accumulator[channel] = 0;

    for (sample=0 ; sample<oversample ; sample++) //Loop to accumulate
        averages
    {

        //Wait for the ADC to be idle
        while (gpio_get_pin_value(ADC_BUSY_PIN));
```

```
    //CS Low
    (&AVR32_GPIO.port[ADC_CS_PIN >> 5])->ovrc = 1 << (ADC_CS_PIN & 0x1F);


    for (channel=0 ; channel<channels ; channel++)
    {
      //Exchange MSB
      (&ADC_USART)->thr = 0xBE;
      while ( !((&ADC_USART)->csr & AVR32_USART_CSR_RXRDY_MASK) );
      rHigh = (&ADC_USART)->rhr;
      //Exchange LSB
      (&ADC_USART)->thr = 0xEF;
      while ( !((&ADC_USART)->csr & AVR32_USART_CSR_RXRDY_MASK) );
      rLow = (&ADC_USART)->rhr;


      accumulator[channel] += ( (rHigh<<8)|(rLow&0xFF) );
    }
    //CS High
    (&AVR32_GPIO.port[ADC_CS_PIN >> 5])->ovrs = 1 << (ADC_CS_PIN & 0x1F);
  }

  for (channel=0 ; channel<channels ; channel++) data[channel] =
    accumulator[channel]/oversample; //Average accumulator and save in
    data[]
}

#endif
```

Clock Code

```
/* adamClocks.h  by   Adam Blake and Zachary Laurencio
   Last modified 2014-06-04
   Function to start up the PLL and transfer the system clock to that
       source.

   **********************************QUICKSTART**********************************

   REQUIRES:    scif_uc3.h
                sysclk.h
                pm_uc3c.h

   SETUP:       Call adamClocks_setup()

   USE:       No further action is required once clocks are started. You're
       good to go with a 48 MHz CPU

   *****************************************************************************
*/

#ifndef ADAMCLOCKS_H_
#define ADAMCLOCKS_H_

#include "scif_uc3c.h"
#include "sysclk.h"
#include "pm_uc3c.h"

//Prototypes
void adamClocks_setup(void);

//Code proper
void adamClocks_setup(void)
```

```c
{
  sysclk_init();
  scif_start_rc8M();
  pm_set_mclk_source(PM_CLK_SRC_RC8M);

  scif_pll_opt_t plloptions =
  {
     .osc         =  SCIF_GCCTRL_RC8M,
     .lockcount   = 16,
     .div         = 1,
     .mul         = 11, //this adds 1
     .pll_div2    = 1,
     .pll_wbwdisable = 0,
     .pll_freq    = 0,
  };

  scif_pll_setup((SCIF_PLL0),&plloptions);
  scif_pll_enable(SCIF_PLL0);
  scif_wait_for_pll_locked(SCIF_PLL0);
  pm_set_mclk_source(PM_CLK_SRC_PLL0);

}

#endif /* ADAMCLOCKS_H_ */
```

Atmel Software Framework Code

```
/**
 * \file
 *
 * \brief Autogenerated API include file for the Atmel Software Framework
    (ASF)
 *
 * Copyright (c) 2012 Atmel Corporation. All rights reserved.
 *
 * \asf_license_start
 *
 * \page License
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions are
    met:
 *
 * 1. Redistributions of source code must retain the above copyright notice,
 *    this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
    notice,
 *    this list of conditions and the following disclaimer in the
    documentation
 *    and/or other materials provided with the distribution.
 *
 * 3. The name of Atmel may not be used to endorse or promote products
    derived
 *    from this software without specific prior written permission.
 *
 * 4. This software may only be redistributed and used in connection with an
 *    Atmel microcontroller product.
```

```
 *
 * THIS SOFTWARE IS PROVIDED BY ATMEL "AS IS" AND ANY EXPRESS OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
 * MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT
     ARE
 * EXPRESSLY AND SPECIFICALLY DISCLAIMED. IN NO EVENT SHALL ATMEL BE LIABLE
     FOR
 * ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
 * ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
 * POSSIBILITY OF SUCH DAMAGE.
 *
 * \asf_license_stop
 *
 */

#ifndef ASF_H
#define ASF_H

/*
 * This file includes all API header files for the selected drivers from
     ASF.
 * Note: There might be duplicate includes required by more than one driver.
 *
 * The file is automatically generated and will be re-written when
 * running the ASF driver selector tool. Any changes will be discarded.
 */

// From module: ADCIFA - ADC Interface A
#include <adcifa.h>
```

```
// From module: Common build items for user board support templates
#include <user_board.h>

// From module: Compiler abstraction layer and code utilities
#include <compiler.h>
#include <status_codes.h>

// From module: FLASHC - Flash Controller
#include <flashc.h>

// From module: GPIO - General-Purpose Input/Output
#include <gpio.h>

// From module: Generic board support
#include <board.h>

// From module: Interrupt management - UC3 implementation
#include <interrupt.h>

// From module: PM Power Manager - UC3 C0/C1/C2 implementation
#include <power_clocks_lib.h>
#include <sleep.h>

// From module: Part identification macros
#include <parts.h>

// From module: SCIF System Control Interface - UC3C implementation
#include <scif_uc3c.h>

// From module: System Clock Control - UC3 C implementation
#include <sysclk.h>
```

```
// From module: USART - Universal Synchronous/Asynchronous
    Receiver/Transmitter
#include <usart.h>


#endif // ASF_H
```

Communication Code

```
/* asynch.h  by   Zachary Laurencio
   Last modified 2014-08-05
   Functions to use usart for sending 16-bit words out for calibration
       purposes.
   Designed to communicate using USART0 in tx only mode.


   ********************************QUICKSTART**********************************


   REQUIRES:     gpio.h
                 usart.h


   SETUP:        Call asynch_setup() to configure USART0


   USE:        Call asynch_word(int16_t data) to transmit a 16-bit word in
       two bytes.


   ****************************************************************************
*/



#ifndef CALIB_H_
#define CALIB_H_

#include "gpio.h"
#include "usart.h"

#define CALIB_USART           AVR32_USART0
#define CALIB_TXD_PIN         AVR32_USART0_TXD_PIN
#define CALIB_TXD_FUNCTION    AVR32_USART0_TXD_FUNCTION

void asynch_setup(void);
```

```c
void asynch_word(int16_t data);


void asynch_setup(void)
{
   gpio_enable_module_pin(CALIB_TXD_PIN,CALIB_TXD_FUNCTION);

   static const usart_options_t calibUsart =
   {
      .baudrate = ASYNCH_BAUDRATE,
      .channelmode = USART_NORMAL_CHMODE,
      .charlength = 8,
      .paritytype = USART_NO_PARITY,
      .stopbits = USART_1_STOPBIT
   };
   usart_init_rs232_tx_only(&(AVR32_USART0),&calibUsart,48000000);
}


void asynch_word(int16_t data)
{
   while (!usart_tx_empty(&CALIB_USART));
   usart_putchar(&CALIB_USART,data>>8);
   while (!usart_tx_empty(&CALIB_USART));
   usart_putchar(&CALIB_USART,data & 0xFF);
}

#endif /* CALIB_H_ */
```

DAC Code

```c
/* dac8581.h  by   Zachary Laurencio
   Last modified 2014-08-03
   Functions to use the DAC8581 16-bit, bipolar DAC.
   Designed to communicate using USART3 in MSPI mode


   *********************************QUICKSTART**********************************


   REQUIRES:     gpio.h
                 usart.h


   SETUP:        Call dac8581_setup() to configure USART3


   USE:       Call dac8581_set(int16_t dacSetting) to set DAC. dacSetting is
       a
              twos-compliment number that sets DAC on range [-Vref , +Vref]


   ****************************************************************************
*/



#ifndef DAC8581_H_
#define DAC8581_H_

#include <avr32/io.h>
#include "board.h"
#include "gpio.h"
#include "usart.h"


#define DAC_USART           AVR32_USART3


#define DAC_CLK_PIN             AVR32_USART3_CLK_3_PIN      //Pin 46
```

```c
#define DAC_CLK_FUNCTION        AVR32_USART3_CLK_3_FUNCTION
#define DAC_TXD_PIN             AVR32_USART3_TXD_2_PIN      //Pin 41
#define DAC_TXD_FUNCTION        AVR32_USART3_TXD_2_FUNCTION
#define DAC_CS_PIN              AVR32_PIN_PC16 //pin 40

//Prototypes
void dac8581_setup(void);
void dac8581_set(int16_t dacSetting);

//Code per se
void dac8581_setup(void)
{

  static const usart_spi_options_t myUsart =
  {
    .baudrate = DAC8581_BAUDRATE,
    .charlength = 8,
    .spimode = 1
  };

  usart_init_spi_master(&(DAC_USART),&myUsart,48000000);

  gpio_enable_module_pin(DAC_TXD_PIN,DAC_TXD_FUNCTION); //configures pins
      for non-GPIO use
  gpio_enable_module_pin(DAC_CLK_PIN,DAC_CLK_FUNCTION);

  gpio_enable_gpio_pin(DAC_CS_PIN);
  gpio_set_pin_high(DAC_CS_PIN);
}

void dac8581_set(int16_t dacSetting)
{
  //CS Low
```

```
//(&AVR32_GPIO.port[DAC_CS_PIN >> 5])->ovrc = 1 << (DAC_CS_PIN & 0x1F);
gpio_clr_gpio_pin(DAC_CS_PIN);


(&DAC_USART)->thr = (dacSetting>>8);
while ( !((&DAC_USART)->csr & AVR32_USART_CSR_TXEMPTY_MASK) );
(&DAC_USART)->thr = (dacSetting & 0xFF);
while ( !((&DAC_USART)->csr & AVR32_USART_CSR_TXEMPTY_MASK) );


//CS High
//(&AVR32_GPIO.port[DAC_CS_PIN >> 5])->ovrs = 1 << (DAC_CS_PIN & 0x1F);
gpio_set_gpio_pin(DAC_CS_PIN);
}




#endif /* DAC8581_H_ */
```

Housekeeping Code

```c
/* hk.h by   Zachary Laurencio
   Last modified 2014-08-07
   Collection of functions to initialize the internal ADC of the
       AT32UC3C2512C for collecting house keeping data.
   Uses the first four ADC inputs.


   *********************************QUICKSTART***********************************


   REQUIRES:     adcifa.h

   GLOBAL VARS: int16_t data    (should be separate from main data array,
       since this will likely be sub-commed into the main array)


   SETUP:        Call hk_setup()


   USE:        Call hk_get_data(int16_t data), where data is the array to be
       populated by conversion results.


   *****************************************************************************

*/


#ifndef HK_H_
#define HK_H_


#include "adcifa.h"


#define V2_PIN        AVR32_ADCIN0_PIN
#define V1_PIN        AVR32_ADCIN1_PIN
#define BATTERY_PIN     AVR32_ADCIN2_PIN
#define TEMP2_PIN     AVR32_ADCIN3_PIN
```

```c
#define TEMP1_PIN     AVR32_ADCIN4_PIN
#define TEMP0_PIN     AVR32_ADCIN5_PIN
#define HK_ADC_FUNCTION  AVR32_ADCIN0_FUNCTION

#define HK_ADC        AVR32_ADCIFA
#define HK_ADC_CONVERSIONS  6

adcifa_sequencer_opt_t mySequencer =
{
   .convnb             = HK_ADC_CONVERSIONS,    // Number of sequences
   .resolution         = ADCIFA_SRES_12B,     // Resolution selection
   .trigger_selection  = ADCIFA_TRGSEL_SOFT,    // Trigger selection
   .start_of_conversion = ADCIFA_SOCB_ALLSEQ,   // Conversion Management
   .sh_mode            = ADCIFA_SH_MODE_OVERSAMP, // Oversampling Management
   .half_word_adjustment = ADCIFA_HWLA_NOADJ,   // Half word Adjustment
   .software_acknowledge = ADCIFA_SA_NO_EOS_SOFTACK // Software Acknowledge
};

//Prototypes
void hk_setup(void);
void hk_get_data(volatile int16_t data[]);

//Code per se
void hk_setup(void)
{
  gpio_enable_module_pin(V2_PIN , HK_ADC_FUNCTION);
  gpio_enable_module_pin(V1_PIN , HK_ADC_FUNCTION);
  gpio_enable_module_pin(BATTERY_PIN , HK_ADC_FUNCTION);
  gpio_enable_module_pin(TEMP2_PIN , HK_ADC_FUNCTION);
  gpio_enable_module_pin(TEMP1_PIN , HK_ADC_FUNCTION);
  gpio_enable_module_pin(TEMP0_PIN , HK_ADC_FUNCTION);

  gpio_enable_module_pin(AVR32_ADCREFP_PIN,AVR32_ADCREFP_FUNCTION);
```

```
gpio_enable_module_pin(AVR32_ADCREFN_PIN,AVR32_ADCREFN_FUNCTION);

adcifa_opt_t myAdcOptions =
{
   .frequency            = 32000,
   .reference_source     = ADCIFA_REF06VDD,
   .sample_and_hold_disable = true,
   .single_sequencer_mode  = false,
   .free_running_mode_enable = false,
   .sleep_mode_enable      = false,

};

adcifa_sequencer_conversion_opt_t
   mySequencerConversionConfig[HK_ADC_CONVERSIONS] =
{
   {
      .channel_p = AVR32_ADCIFA_INP_ADCIN0, // Positive Channel
      .channel_n = AVR32_ADCIFA_INN_GNDANA, // Negative Channel
      .gain    = ADCIFA_SHG_1        // Gain of the conversion
   },
   {
      .channel_p = AVR32_ADCIFA_INP_ADCIN1, // Positive Channel
      .channel_n = AVR32_ADCIFA_INN_GNDANA, // Negative Channel
      .gain    = ADCIFA_SHG_1        // Gain of the conversion
   },
   {
      .channel_p = AVR32_ADCIFA_INP_ADCIN2, // Positive Channel
      .channel_n = AVR32_ADCIFA_INN_GNDANA, // Negative Channel
      .gain    = ADCIFA_SHG_1        // Gain of the conversion
   },
   {
      .channel_p = AVR32_ADCIFA_INP_ADCIN3, // Positive Channel
```

```c
        .channel_n = AVR32_ADCIFA_INN_GNDANA, // Negative Channel
        .gain      = ADCIFA_SHG_1           // Gain of the conversion
      },
      {
        .channel_p = AVR32_ADCIFA_INP_ADCIN4, // Positive Channel
        .channel_n = AVR32_ADCIFA_INN_GNDANA, // Negative Channel
        .gain      = ADCIFA_SHG_1           // Gain of the conversion
      },
      {
        .channel_p = AVR32_ADCIFA_INP_ADCIN5, // Positive Channel
        .channel_n = AVR32_ADCIFA_INN_GNDANA, // Negative Channel
        .gain      = ADCIFA_SHG_1           // Gain of the conversion
      }
    };

  adcifa_configure(&HK_ADC , &myAdcOptions , 48000000);
  adcifa_configure_sequencer(&HK_ADC , 0 , &mySequencer ,
     mySequencerConversionConfig);

}


void hk_get_data(volatile int16_t data[])
{
  adcifa_start_sequencer(&HK_ADC, 0);
  adcifa_get_values_from_sequencer(&HK_ADC , 0 , &mySequencer , data);
}

#endif /* HOUSEKEEPING_H_ */
```

MTeX Define Code

```
#ifndef MTEX_H_
#define MTEX_H_

#define MDCPP_CHANNELS 6
#define MDCPO_CHANNELS 3
#define MDCPN_CHANNELS 6
#define ACC_CHANNELS 3
#define SLP_CHANNELS 2

#define MDCPP_OVER    4
#define MDCPO_OVER    6
#define MDCPN_OVER    4
#define ACC_OVER   6
#define SLP_OVER   2

#define MDCPP_TAG  0x9000
#define MDCPO_TAG  0x9100
#define MDCPN_TAG  0x9200
#define ACC_TAG       0x9300
#define SLP_TAG       0x9400
#define WADIS_TAG  0x9500

#define TAG_EXP       0x0000
#define TAG_01     0x0010
#define TAG_02     0x0020
#define TAG_03     0x0030
#define TAG_04     0x0040
#define TAG_05     0x0050
#define TAG_06     0x0060
#define TAG_07     0x0070
#define TAG_08     0x0080
```

```
#define TAG_09      0x0090
#define TAG_0A      0x00A0


const uint8_t hkOrder[] = {2, 4, 5, 3, 1, 0}; //BATT, INT1, INT2, EXT,
    V1(5), V2(12)


#endif /* MTEX_H_ */
```

5.4   Matlab Codes

5.4.1   SLP Calibration Code

Calibration Codes

```
function data = SLP_Calibration_Manual(serialNumber, distance, temp)
    delete(instrfind);
    baudrate = 115200;  %symbols per second of the com port
    packetLength = 50000; % 2 x (2 + 2x2x396 + 6) + spare

    synch1 = 204;  %CC
    synch2 = 170;  %AA
    synch3 = 227;  %E3
    synch4 = 140;  %8C
    synchWord = -13142;
    channels = 2;

    file =
        sprintf('SLPtest%02dDISTANCE%03dTEMP%03d.dat',serialNumber,distance,temp+273);


    clc

    s = serial('COM5');                              %Create a serial object
```

```matlab
set(s,'baudrate',baudrate);              %Set baudrate
set(s,'inputbuffersize',1000000);

data =[];
out = [];
byteData =[];
wordData = [];

%% Collect data
fopen(s);                                %Open port
while (s.BytesAvailable < packetLength)
end
out = fread(s,s.BytesAvailable)';
fclose(s);
delete(s);

%% Frame synch
while ( (numel(out)>10) && ( (out(1) ~= synch1) || (out(2)~=synch2) ||
    (out(3)~=synch3) || (out(4)~=synch4) ) )
    out(1) = 0;
    out = out(find(out == synch1,1,'first'):numel(out));
end

%% Convert to words
if (numel(out) > (channels+1)*(2))
    byteData = out(1:numel(out));
    for i=1:floor(numel(byteData)/2)
        wordData(i) = byteData(2*i-1)*256+byteData(2*i);
        if (wordData(i) >= 2^15)
            wordData(i) = wordData(i)-2^16;
        end

    end
```

```matlab
    end

    %% Trim to just data (no framesynchs)
    trim = find(wordData == synchWord,2);
    trim = trim(2);
    wordData = wordData(3:trim-1);

    %% Save data to file
    f = fopen(file,'w');
    fprintf(f,'%d\t',wordData);
    fclose(f);

    %% flash a plot to evaluate data
    p = plot(wordData,'+');
    pause(0.3);
%    close all;

    %% output data to MATLAB for easy checking
    data = wordData;
    s = sprintf('%s Saved...',file);
    disp(s)

end
```

DAC Calibration Code

```matlab
clc
clear all
delete(instrfind)


file = fopen('test.txt','w');


for i=1:1:716
    sweepvalues(i,1) = 65*(i-1)-14700 ;
end


% For board
r = serial('COM5');
set(r, 'BaudRate', 115200);
fopen(r);


%For multimeter
t = serial('COM15');
set(t, 'BaudRate', 38400);
fopen(t);


Buffer = [];


for h=1:5

    % reset dac to zero
    fprintf(r, 's');

    for i=1:716
    % while(1)
    %     Value the dac is at
        sweepvalues(i,1) = 65*(i-1)-14700 ;
```

```matlab
%    send command to measure
    fprintf(t, 'MEAS?');
    voltage = fgetl(t);
    extra = fgetl(t);
%    Read in housekeeping
    a = fread(r,12,'uint8');
    flushinput(r);
    DataArrayUint8 = a();
    %converting int8 to int16
    for Row = 1:size(DataArrayUint8,1)/2
        DataArrayUint16(Row,1) =
            DataArrayUint8(2*Row-1)*256+DataArrayUint8(2*Row);
    end
%   record DAC voltage
%    fprintf(file,'Voltage is %s at a DAC count of
    %d\n',voltage,sweepvalues(i,1));
    fprintf(file,'%s %d %d %d %d %d %d \n',voltage,sweepvalues
(i,1),DataArrayUint16(1,1),DataArrayUint16(2,1),DataArrayUint16(3,1),
DataArrayUint16(4,1),DataArrayUint16(5,1),DataArrayUint16(6,1));
%   step the DAC
    fprintf(r, 'd');
%    pause(0.01);
end


    fprintf(r, 'w');
%    waste = fread(r,12,'uint8');
%    pause(0.01);
for i=1:716
%    Value the dac is at
    sweepvalues(i+716,1) = 31775 - 65*(i-1);
%    send command to measure
    fprintf(t, 'MEAS?');
```

```matlab
        voltage = fgetl(t);
        extra = fgetl(t);
%     Read in housekeeping
        a = fread(r,12,'uint8');
        flushinput(r);
        DataArrayUint8 = a();
        %converting int8 to int16
        for Row = 1:size(DataArrayUint8,1)/2
            DataArrayUint16(Row,1) =
                DataArrayUint8(2*Row-1)*256+DataArrayUint8(2*Row);
        end
%   record DAC voltage
%     fprintf(file,'Voltage is %s at a DAC count of
        %d\n',voltage,sweepvalues(i,1));
        fprintf(file,'%s %d %d %d %d %d %d \n',voltage,sweepvalues
(i+716,1),DataArrayUint16(1,1),DataArrayUint16(2,1),DataArrayUint16(3,1),
DataArrayUint16(4,1),DataArrayUint16(5,1),DataArrayUint16(6,1));
%   step the DAC
        fprintf(r, 'a');
%     pause(0.01);
    end

end
```

Data Plotting

```
close all
clear all
clc

A1 = load('SLP00DISTANCE82.50TRIAL030CURRENT11.50.dat');
A1up = A1(1,1:1312)';
A1down = A1(1,1316:2627)';
A1HG(:,1) = A1up(2:2:end,1);
A1HG(:,2) = flipud(A1down(2:2:end,1));
A1LG(:,1) = A1up(1:2:end,1);
A1LG(:,2) = flipud(A1down(1:2:end,1));


A2 = load('SLP00DISTANCE82.50TRIAL031CURRENT11.50.dat');
A2up = A2(1,1:1312)';
A2down = A2(1,1316:2627)';
A2HG(:,1) = A2up(2:2:end,1);
A2HG(:,2) = flipud(A2down(2:2:end,1));
A2LG(:,1) = A2up(1:2:end,1);
A2LG(:,2) = flipud(A2down(1:2:end,1));


A3 = load('SLP00DISTANCE82.50TRIAL032CURRENT11.50.dat');
A3up = A3(1,1:1312)';
A3down = A3(1,1316:2627)';
A3HG(:,1) = A3up(2:2:end,1);
A3HG(:,2) = flipud(A3down(2:2:end,1));
A3LG(:,1) = A3up(1:2:end,1);
A3LG(:,2) = flipud(A3down(1:2:end,1));



x=1:656;
p = plot(x,A1LG,'.k',x,A2LG,'.b',x,A3LG,'.g');
```

```matlab
legend(p(1:2:end),'25.5 Hz Sweep','8.26 Hz Sweep','0.85 Hz
    Sweep','location','Northwest');
% xlim([450 656])
xlabel('Steps')
ylabel('Counts')
```

5.4.2 I-V Codes

Retardation Code

```matlab
clear all
close all
clc

load('LowGain.mat')
load('HighGain.mat')

distances = [135 150 160 170 180 190 200 210 220 230 240 250 260 270 280
    290 300 310 320 330 340 350 360 370 380 390 400 410 420 430 440 450 460
    470 480 490 500]';

%convert voltages
for i=1:2621
    Voltage(i,1) = ((i-1)*25-32750)*(5/(2^15));
end
%convert currents
LowGainCurrent = lowGain(1:2621,:)*5/((2^15)*2*1000);
HighGainCurrent = highGain(1:2621,:)*5/((2^15)*20*1000);
%log for retardation region
LogLowGainCurrent = log(LowGainCurrent);
LogHighGainCurrent = log(HighGainCurrent);

% Finding cuts
[M,I] = max(abs(LogLowGainCurrent));
cutLowLow = I';
[M,I] = max(diff(LowGainCurrent));
cutLowHigh = I';

toperror = 0.3;
bottomerror = 0.3;
```

```matlab
% for low gain
for j=1:size(LowGainCurrent,2)
    difference = cutLowHigh(j) - cutLowLow(j);

    LowGainRetardationCurrent = LogLowGainCurrent((cutLowLow(j)+floor
      (difference*bottomerror)):(cutLowHigh(j)-ceil(difference*toperror)),j);
    RetardationVoltage = Voltage((cutLowLow(j)+floor(difference*bottomerror))
      :(cutLowHigh(j)-ceil(difference*toperror)),:);
    p = polyfit(RetardationVoltage,LowGainRetardationCurrent,1);
    slope(j,1) = p(1);
    Temp_Kelvin(j,1) = (1.602*10^-19 / (1.38*10^-23 * slope(j)));
    Temp_eV(j,1) = Temp_Kelvin(j)/(1.16*10^4);
    figure;
    plot(RetardationVoltage,LowGainRetardationCurrent,'.b');
    hold on;
    plot(RetardationVoltage,polyval(p,RetardationVoltage),'.g');
    set(gcf,'color','w');
    xlabel('Sweep Voltage (Volts)')
    ylabel('Ln(Current)')
    legend('Data','Linear Fit','Loctaion','NorthEast')
%

end

% Finding cuthigh
[M,I] = max(abs(LogHighGainCurrent));
cutHighLow = I';
[M,I] = max(diff(HighGainCurrent));
cutHighHigh = I';

% for high gain
for j=1:size(HighGainCurrent,2)
```

```matlab
        difference = cutHighHigh(j) - cutHighLow(j);

        HighGainRetardationCurrent = LogHighGainCurrent((cutHighLow(j)+floor
          (difference*bottomerror)):(cutHighHigh(j)-ceil(difference*toperror)),j);
        RetardationVoltage =
            Voltage((cutHighLow(j)+floor(difference*bottomerror))
          :(cutHighHigh(j)-ceil(difference*toperror)),:);
        p = polyfit(RetardationVoltage,HighGainRetardationCurrent,1);
        slope(j,2) = p(1);
        Temp_Kelvin(j,2) = (1.602*10^-19 / (1.38*10^-23 * slope(j,2)));
        Temp_eV(j,2) = Temp_Kelvin(j,2)/(1.16*10^4);
%     plot(RetardationVoltage(),LowGainRetardationCurrent())
%     set(gcf,'color','w');
%     xlabel('Sweep Voltage (Volts)')
%     ylabel('Ln(Current)')


end

Temp(:,1) = Temp_Kelvin(1:2:end,1);
Temp(:,2) = Temp_Kelvin(2:2:end,1);
Temp(:,3) = Temp_Kelvin(1:2:end,2);
Temp(:,4) = Temp_Kelvin(2:2:end,2);
plot(distances,Temp,'.')
set(gcf,'color','w');
legend('Low Gain Up Sweep','Low Gain Down Sweep','High Gain Up Sweep',
    'High Gain Down Sweep')
ylabel('Temperature (Kelvin)')
xlabel('Chamber Distance (mm)')
```

Electron Saturation Code

```
clear all
close all
clc

load('LowGain.mat')
load('HighGain.mat')

%convert voltages
for i=1:2621
    Voltage(i,1) = ((i-1)*25-32750)*(5/(2^15));
end
%convert currents
LowGainCurrent = lowGain(1:2621,:)*5/((2^15)*2*1000);
%log
LogLowGainCurrent = log(LowGainCurrent);

% Finding cut
[M,I] = max(diff(LowGainCurrent));
cutLow = I';
PlasmaPotential = cutLow;


bottomerror = 0.2;

% for low gain
for j=1:5%size(LowGainCurrent,2)
    difference = size(LowGainCurrent,1)-cutLow(j);

    LowGainRetardationCurrent =
        LowGainCurrent(floor(cutLow(j)+difference*bottomerror):end,j).^2;
    RetardationVoltage =
        Voltage(floor(cutLow(j)+difference*bottomerror):end,:);
```

```matlab
    p = polyfit(RetardationVoltage,LowGainRetardationCurrent,1);
    fit(j,1) = p(1);
    fit(j,2) = p(2);
    top = sqrt((fit(j,1)*2*pi*9.109e-31)/(1.602e-19));
    bottom = ((1.602e-19)*2*pi*0.03175*0.03175/4);
    ElectronDensity(j,1) = top/bottom;
    Temp_Kelvin(j,1) = (fit(j,2)*9.109e-31)/((1.602e-19)^2*
      (ElectronDensity(j,1))^2*2*pi*(0.03175/2)^4*1.38e-23);
    Temp_eV(j,1) = Temp_Kelvin(j)/(1.16*10^4);
    figure;
    plot(RetardationVoltage,LowGainRetardationCurrent,'.b');
    hold on;
    plot(RetardationVoltage,polyval(p,RetardationVoltage),'.g');
    set(gcf,'color','w');
    xlabel('Sweep Voltage (Volts)')
    ylabel('Squared Saturation Current (A^2)')
end

plot(ElectronDensity)
set(gcf,'color','w');
xlabel('Sweep Number')
ylabel('Electron Density (1/m^3)')




% plot(Temp_Kelvin)
% legend('Low Gain','High Gain')
% set(gcf,'color','w');
% % xlabel('')
% ylabel('Temperature (Kelvin)')
```

Ion Saturation Code

```matlab
clear all
close all
clc

load('HighGain.mat')

%convert voltages
for i=1:2621
    Voltage(i,1) = ((i-1)*25-32750)*(5/(2^15));
end
%convert currents
HighGainCurrent = highGain(1:2621,:)*5/((2^15)*2*1000);
%log
LogHighGainCurrent = log(HighGainCurrent);


% Finding cut
[M,I] = max(abs(log(HighGainCurrent)));
cutHigh = I';
PlasmaPotential = cutHigh;


toperror = 0.0;


% for low gain
% j=1;
for j=1:10%size(LowGainCurrent,2)
    difference = size(HighGainCurrent,1)-cutHigh(j);



    HighGainSaturationCurrent = HighGainCurrent(1:floor(cutHigh -
        toperror*difference),j);
    SaturationVoltage = Voltage(1:floor(cutHigh - toperror*difference),:);
```

```matlab
    p = polyfit(SaturationVoltage,HighGainSaturationCurrent,1);
    fit(j,1) = p(1);
    fit(j,2) = p(2);

    figure;
    plot(SaturationVoltage,HighGainSaturationCurrent,'.b');
    hold on;
    plot(SaturationVoltage,polyval(p,SaturationVoltage),'.g');
    set(gcf,'color','w');
    xlabel('Sweep Voltage (Volts)')
    ylabel('Squared Saturation Current (A^2)')

%   its a better line fit for I^2
    HighGainSaturationCurrent = HighGainSaturationCurrent.^2;
    p = polyfit(SaturationVoltage,HighGainSaturationCurrent,1);
    fit(j,1) = p(1);
    fit(j,2) = p(2);

    figure;
    plot(SaturationVoltage,HighGainSaturationCurrent,'.b');
    hold on;
    plot(SaturationVoltage,polyval(p,SaturationVoltage),'.g');
    set(gcf,'color','w');
    xlabel('Sweep Voltage (Volts)')
    ylabel('Squared Saturation Current (A^2)')


% equation for plasma density is in dr b;s thesis on page 39


end


% plot(ElectronDensity)
```

```
% set(gcf,'color','w');
% xlabel('Sweep Number')
% ylabel('Electron Density (1/m^3)')
```