DEVELOPMENT OF THE DIGITAL SIGNAL PROCESSING FOR THE SPACE WEATHER PROBES VERSION 2 SENSOR USING THE MATLAB/SIMULINK ENVIRONMENT

by

Benjamin J. Lewis

A thesis submitted in partial fulfillment
of the requirements for the degree

of

MASTER OF SCIENCE

in

Electrical Engineering

Approved:

Charles Swenson, Ph.D. Major Professor	Todd Moon, Ph.D. Committee Member
Randy Jost, Ph.D. Committee Member	D. Richard Cutler, Ph.D. Vice Provost of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2023

ABSTRACT

Development of the Digital Signal Processing for the Space Weather Probes Version 2 Sensor Using the MATLAB/Simulink Environment

by

Benjamin J. Lewis, Master of Science

Utah State University, 2023

Major Professor: Charles Swenson, Ph.D.
Department: Electrical and Computer Engineering

Ionospheric probes provide measurements that are the basis for building predictive models of the ionosphere. These ionospheric predictions are important for studying ionospheric phenomena, such as plasma bubbles. These phenomena can interfere with radio communications between satellites and the ground.

Space Weather Probes (SWP) is an instrument developed by Utah State University that provides measurements of the ionospheric plasma. SWP was flown on the Scintillation Prediction Observation Task (SPORT) mission, a joint mission between the United States of America and Brazil. This thesis will develop and test the digital signal processing (DSP) hardware design for the Space Weather Probes version 2 (SWP2). The SWP2 instrument suite consists of the Floating Potential Probe (FPP), the Electric Field Wave Spectrometer (EFW), the Fixed Bias Langmuir Probe (FLP), the Sweeping Langmuir Probe (SLP), and the Sweeping Impedance Probe (SIP).

The data from SWP2 will be used to determine the density and temperature of the local plasma, as well as the electric field in the local plasma. Spectrograms of the power spectral density of the electric field and plasma density will be measured. The Sweeping Impedance Probe will be controlled with a software defined radio. The Sweeping Impedance

Probe also has functionality to track the resonant peak of the plasma at the upper hybrid frequency and make measurements of the quality factor of the peak. All DSP designs are implemented using the MATLAB/Simulink HDL coder tool.

(100 pages)

PUBLIC ABSTRACT

Development of the Digital Signal Processing for the Space Weather Probes Version 2 Sensor Using the MATLAB/Simulink Environment

Benjamin J. Lewis

Space Weather Probes (SWP) is an instrument that provides measurements of the plasma environment of the ionosphere. SWP was flown on the Scintillation Prediction Observation Task (SPORT) mission, a joint mission between the United States of America and Brazil. This thesis will develop the digital signal processing (DSP) hardware design for the Space Weather Probes version 2 (SWP2). The data from these instruments will be used to determine the density and temperature of the local plasma, as well as the electric field in the local plasma. This thesis presents the design and testing of the DSP designs for all of the different probes in SWP2. All design and testing was done in the MATLAB/Simulink environment.

CONTENTS

LIST OF FIGURES

ACRONYMS

ADC	Analog to Digital Converter
DAC	Digital to Analog Converter
DSP	Digital Signal Processing
EFP	Electric Field Probe
FBP	Fixed Bias Probe
FFT	Fast Fourier Transform
HDL	Hardware Description Language
ISS	International Space Station
FPGA	Field Programmable Gate Array
FPMU	Floating Potential Measurement Unit
LPF	Low Pass Filter
NCO	Numerically Controlled Oscillator
PLL	Phase-Locked Loop
PSD	Power Spectral Density
RF	Radio Frequency
SDR	Software Defined Radio
SIP	Sweeping Impedance Probe
SLP	Sweeping Langmuir Probe
SPORT	Scintillation Prediction Observations Research Task
SWP	Space Weather Probes
SWP2	Space Weather Probes version 2
USU	Utah State University
VHDL	VHSIC Hardware Description Language
VHSIC	Very High Speed Integrated Circuit

CHAPTER 1

Introduction

Microprocessors, transistors, and logic gates are the fundamental components that drive the modern digital world. Computer microprocessors are comprised of millions of transistors that are arranged into digital circuits called logic gates. These digital logic gates are the building blocks of microprocessors or any digital computing device. The digital logic gates are arranged to perform a basic set of mathematical operations or into very complex computing structures, or processors, that are controlled by programs. A program is a list of instructions that informs the digital logic processor what operations to perform and in what order. Programs can be written that implement complex algorithms and operations on microprocessors. Microprocessors are commonly used in small satellites to control the orientation of the spacecraft and the flow of data between sensors and the communication radios. The signals from sensors are calibrated, cleaned, and prepared for used in a process called signal processing by microprocessors.

An alternative to microprocessors is custom hardware. Custom hardware refers to specialized digital circuitry that performs a specific task and only that task. They cannot be reprogramed by software, but the tradeoff is much greater speed and energy efficiently for the calculation or signal processing task. A common method of building custom hardware is by using Field-Programmable Gate Arrays (FPGA). An FPGA is a semiconductor device that contains an array of unconnected digital logic element called gates and logic blocks. These basic elements can be interconnected to form specific digital computing functions through a re-programmable configuring process. Using these building blocks, complex signal processing algorithms can be implemented on the FPGA like a microprocessor but more efficiently. The FPGA can be reconfigured by uploading a new specification for the interconnections of its gates and logic blocks to implement different custom hardware designs. Due to the customizable hardware of FPGAs, FPGAs excel at parallel processing and real-time

applications. An FPGA can be configured to perform multiple signal processing chains in parallel. The hardware of the FPGA is explicitly defined, so the timing of the hardware is precise and constant. This allows for precise timing control which is essential for real-time processing.

FPGAs are typically programmed using a hardware description language (HDL) such as Verilog or VHDL. Developing in these languages is like programing a microprocessor but requires expertise in low-level design of digital logic. This complexity is a barrier to the wide use of FPGA's for signal processing because the description of the interconnections withing the FPGA can be time-consuming and error prone. Recently new tools have been developed to simplify the development of HDL code using high-level graphical description languages and large libraries of standard computing functions. One example developed by the MathWorks company is a set of tools called MATLAB, Simulink, and HDL Coder. These tools allow for the high-level description of FPGA design. The graphical development environment of Simulink is used to show the interconnection of digital logic elements for a FPGA in the form of block diagrams. The block diagrams are then converted to HDL code using the HDL Coder tool. The standard tools for FPGA design, including synthesis, placement, routing, and configuration are used to create a programing file for the FPGA to create a custom hardware design of a digital signal process algorithm or control function.

This thesis explores using FPGA based custom hardware to perform signal processing for science instruments studying space weather. Space weather is the conditions of the upper atmosphere, ionosphere, and the region surrounding the Earth. Space weather is primarily caused by the interactions between solar activity and the Earth's magnetic field. The ionosphere of the earth consists of plasma, a mix of neutral gasses, electrons and ions.

Utah State University produced a set of in-situ ionospheric diagnostic instruments called Space Weather Probes (SWP) for the CubeSat mission named SPORT. SPORT is a joint mission between the United States and Brazil to investigate a space weather phenomena called plasma bubbles. NASA provided the science instruments, including the USU SWP, and the launch of SPORT. Brazil provided the spacecraft and combined it

with the NASA provided instruments and the operations of the spacecraft. The goal of the mission is to investigate the conditions in the ionosphere that lead to the formation of plasma bubbles. Plasma bubbles are large structures of plasma density depleation that form in the ionosphere near the equator and after sunset. The bubble upwards from the base of the ionosphere at 90 km and sometimes reach altitudes of 1000 km. The USU SWP measures the temperature and density of the ionosphere to help understand the conditions that give rise to plasma bubbles. SPORT has been in orbit since December 2022.

This thesis describes the development of the FPGA based custom hardware and the digital signal processing for the second generation of the USU Space Weather Probes (SWP2). The SWP sensor package makes use of an FPGA which is programed using the MATLAB/Simulink HDL Coder tool chain. SWP2 includes five probes, a floating potential probe (FPP), an electric field wave spectrometer (EFW), a fixed Langmuir probe (FLP), a sweeping Langmuir probe (SLP), and an impedance probe (IP). This redesign of the SWP is based on the lessons learned and to fix errors found in the SWP built for the SPORT mission [3, 6]. The new design approach streamlines the development of the firmware for SWP2, optimize signal filtering, and add new functionality to the probes.

1.1 Research Objectives

The objective of this thesis is to address the following questions:

1. Can the MathWorks MATLAB/Simulink environment be used to develop the DSP system for SWP2 in an effective way that can be maintained at USU?

2. Can the MathWorks MATLAB/Simulink environment be used to effectively verify the performance of the SWP2 DSP design before implementation on a FPGA?

3. Can additional functionality be added to the impedance probe to measure the quality factor of the upper hybrid resonance in the characteristics of the probe-plasma impedance?

1.1.1 Can the MathWorks MATLAB/Simulink environment be used to develop the DSP system for SWP2 in an effective way that can be maintained at USU?

High level design tools, such as Simulink and HDL Coder, have been suggested as a way to overcome challenges of FPGA design and accelerate FPGA hardware development. Simulink provides a library of signal processing blocks, as well as other versatile control block, such as state machines. The signal processing blocks are used to develop the various algorithms used in the design. Several pre-built Simulink DSP blocks are planned to be used in the development of the SWP2 including:

- Fast Fourier Transform (FFT)
- Filter Downsample
- Numerically Controlled Oscillator (NCO)
- Four Quadrant Arctangent (Atan2)

1.1.2 Can the MathWorks MATLAB/Simulink environment be used to effectively verify the performance of the SWP2 DSP design before implementation on a FPGA?

When creating digital designs for an FPGA, it is essential to verify that the design has the desired function. Using the MATLAB/Simulink environment to create expected inputs and examine outputs is a useful way to verify the design and streamlines the FPGA development process. The simulation tools provide logging of any signal withing the design which allows behavioral verification before it is placed on the FPGA and such internal signals are inaccessible. The Simulink environment allows for signals to be logged to files from long runs and then examined to find the causes of unexpected or undesired behavior. Final verification tests to show compliance with required behavior can be created by injecting the prototype FPGA design with pre-defined signals and demonstrating the required behavior of the FPGA.

1.1.3 Can additional functionality be added to the impedance probe to measure the quality factor of the upper hybrid resonance in the characteristics of the probe-plasma impedance?

The impedance probe measures the impedance of a probe in a plasma as a function of frequency. The impedance has resonant conditions with quality factors (Q-factor) that have not been measured in detail. The basic comparison of theory to observations indicates that the quality of the resonance is not well predicted. This thesis develops a method of tracking the resonant condition and measuring the Q-factor of the plasma impedance.

1.2 Background

Earth's ionosphere consists of ionized gases called a plasma that extends upwards starting at around 50 km above sea level [1]. The ionosphere is divided into multiple layers as shown in Figure 1.1. The mean densities and altitudes of these layers varies with the time of day, latitude, and longitude. The ionosphere also experiences hourly and daily changes or disturbances that are akin to weather. Some ionospheric disturbances can be traced back to the sun and are caused by solar flares, coronal mass ejections, and sunspot activity while others are connected to tropospheric events like thunderstorms, mountain ranges, and hurricanes.

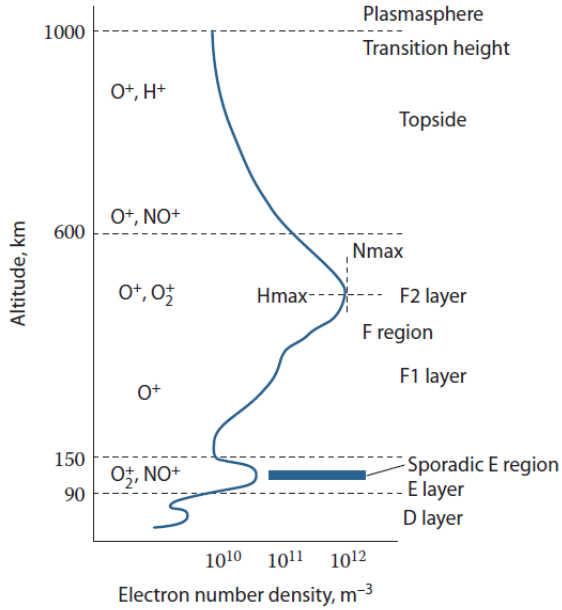

Fig. 1.1: Daytime ionospheric layers [1]

The electron density of the plasma is the primary parameter of the ionosphere because of the electron's low mass and high mobility. Due to the population of freely moving electrons in the ionosphere, the ionosphere can interact with electromagnetic radio waves. Radio waves are reflected off the ionosphere if their frequency is at or below a critical frequency for the ionosphere. The critical frequency is related to the electron density of the ionosphere as seen in equation 1.1 [1] where n is the electron density, q_e is the charge of an electron, m_e is the mass of an electron, and ϵ_0 is the permittivity of free space. As the electron density increases, the critical frequency increases. Above the critical frequency, the ionosphere becomes transparent to radio waves. Knowing the critical frequency of the ionosphere, or the electron density, is critical to understanding the effects of the ionosphere on communication, GPS navigation, and other radio systems.

$$f_p = \frac{1}{2\pi}\sqrt{\frac{nq_e^2}{m_e\epsilon_0}} \tag{1.1}$$

Other key ionospheric plasma parameters include the electron temperature and the local electric field. In the Earth's magnetic field, the motion, or winds, of the ionospheric

plasma are accompanied by an electric field that allows the plasma to cross magnetic field lines. These plasma parameters, density, temperature, and motion are measured by SWP2. The electron density is measured using two different methods called the Langmuir and impedance probe techniques. The Langmuir probe operates by making DC measurements of currents collected from the plasma environment from which the electron density and temperature are determined. The impedance probe makes AC measurements at radio frequencies of the currents collected from the plasma environment from which the electron density is determined. The DC and high frequency electric fields is measured by observing the voltage between two sensors separated from each other called a floating potential probe. The high frequency electric fields are quantified using a spectrometer.

1.2.1 Langmuir Probe Theory

The Langmuir probe functions by placing a conductor in a plasma. The conductor is given a bias voltage and the current collected by the conductor is measured. This creates a current voltage (IV) curve as seen in figure 1.2. This curve is divided into three regions. When the bias voltage is low, the probe is in the ion saturation region. In this region, the probe is attracting positive ions from the plasma. When the bias voltage reaches the floating potential voltage of the plasma the probe enters the electron retardation region. In this region the probe begins collecting the electrons in the plasma. The electron retardation region is exponential, and the e-folding scale is determined by the temperature of the plasma. When the bias voltage crosses the plasma potential, the probe enters the electron saturation region. In this region the probe is attracting electrons. The shape of this region is dependent on the geometry of the probe. The number of electrons collected in this region, the current, is proportional to the electron density.

The SWP2 instrumentation includes two different Langmuir probes called the sweeping Langmuir probe (SLP) and the fixed bias Langmuir probe (FLP). The SLP sweeps the probe voltage to measure the IV curve concentrating on the electron retardation region. Both the electron density and temperature are determined from each SLP voltage sweep. The FLP is used to as a constant monitor of collection current where the voltage level is chosen to

be in the electron saturation region. The FLP measures the relative changes in electron density.

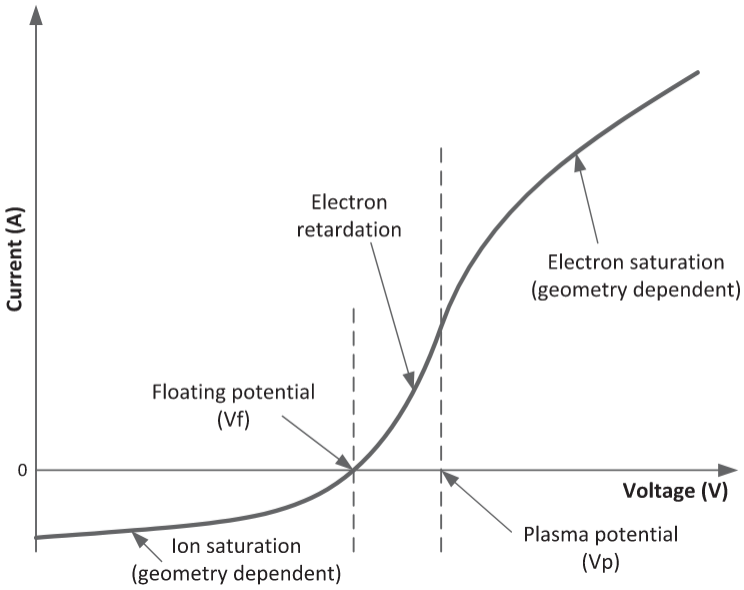

Fig. 1.2: Sample IV Curve for a Cylindrical Langmuir Probe [2]

The current collected from the ionosphere, when using the Langmuir technique, is proportional to a product of the saturation current, J_{sat_j} , the probe area, A, and a collection factor, $F(\Phi)$, which is a function of the normalized sheath potential, Φ_s for each species j with charge, q_j, temperature T_j and mass m_j as given by equation 1.2

$$I = \sum_j J_{sat_j} A F(\Phi_{s_j}),\ where\ J_{sat_j} \equiv q_j N_j \sqrt{\frac{k_b T_j}{2\pi m_j}}\ and\ \Phi_{s_j} \equiv \frac{-q_j \phi_s}{k_b T_j} \tag{1.2}$$

Analytic expressions for $F(\Phi)$, for different shaped bodies that are both stationary and moving at orbital velocities, have been developed by many including Mott-Smith and Langmuir [7], Kanal [8], Hoegy and Warton [9,10], Laframboise [11,12], Allan [13], Chen [14,15], and extensively studied along with the plasma sheaths that form around bodies in the space environment. Approximations for the collection factor for several probe geometries is shown in equation 1.3.

$$F(\Phi_{s_j}) = \begin{cases} \exp(\Phi_{s_j}), & \Phi_{s_j} < 0 \\ (1+\Phi_{s_j})^{\beta}, & \Phi_{s_j} \geq 0 \end{cases}, \quad \beta = \begin{cases} 1, & \text{spherical} \\ \frac{1}{2}, & \text{cylindrical} \\ 0, & \text{planar} \end{cases} \tag{1.3}$$

It is important to recognize the difference between ϕ_s, the voltage across the plasma sheath enveloping the collecting area, and the bias voltage, V_b, applied to the sensing surface by the electronics of the measurement instrumentation. The bias voltage is necessarily referenced to the ground of the spacecraft power system which is tied to the spacecraft structure. The current collected from the ionosphere using the Langmuir technique flows to the sensor surface, through the instrumentation to the ground of the power system and is then returned to the ionosphere via the surface of the spacecraft. The sheath potential of the collecting area is $\phi_s \approx V_b + \phi_{sc}$ where ϕ_{sc} is the potential of the spacecraft relative to the undisturbed ionosphere, or spacecraft charging. Thus, the entire surface of the spacecraft becomes an integral part of the measurement. When the surface area of the spacecraft is 1000 times larger than the Langmuir sensor's surface, as was historically typical for large spacecraft, then ϕ_{sc} is approximately constant [16]. But on smaller buses like SPORT and ACMES (e.g., CubeSats and μSats), the spacecraft charge will be driven by the exposed Langmuir sensor bias [2, 17, 18]. This creates problem, as an unknown value for ϕ_{sc} can be suitably determined using a Langmuir voltage sweep only when it is not a function of the sweep voltage. The effects of an unknown value for ϕ_{sc} can be corrected by measuring the floating potontial with a floating potential probe as seen in figure 1.3. The sheath potential of the collecting area is calculated using equation 1.4. Thus, SWP2 will actively and in real-time monitor the change in the spacecraft potential by recording the floating potential of a Cube Sat spacecraft reference surface at every step of the Langmuir sweep with the FPP.

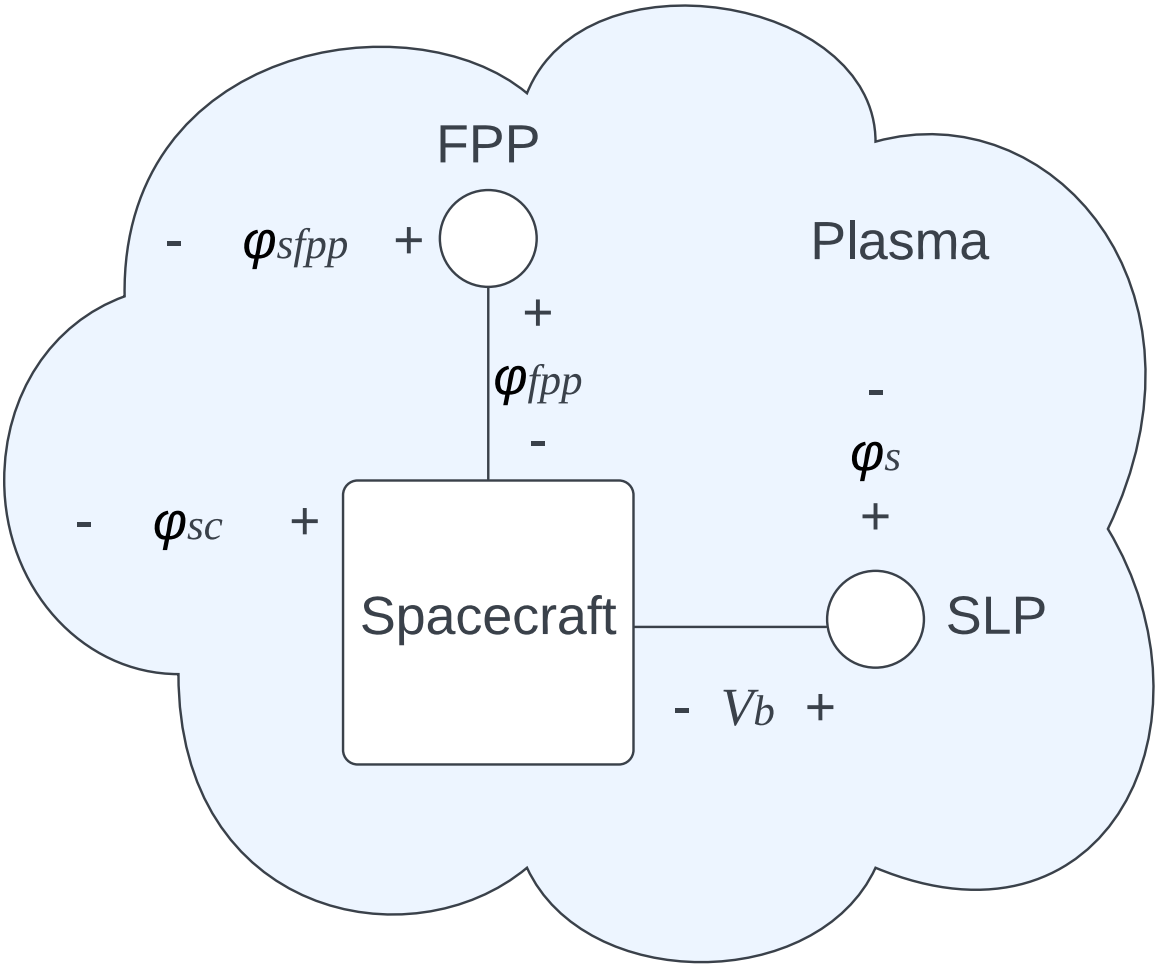

Fig. 1.3: SLP and FPP Probe Potentials

$$\phi_s = V_b + \phi_{sfpp} - \phi_{fpp} \tag{1.4}$$

1.2.2 Impedance Probe Theory

The impedance probe functions as the measurement of a capacitor where the dielectric within the capacitor is the ionosphere (figure 1.4), which significantly changes its characteristics. The impedance of the probe's capacitance is measured over a range of RF frequencies by applying a voltage signal with a known amplitude and frequency to the probe as described by equation 1.5. The current signal passing through the probe's capacitance is measured in magnitude and phase and used to calculate the impedance using equation 1.6.

The signal is swept over a range of RF (1-30 MHz) frequencies resulting in a impedance verses frequency curve as shown in figure 1.5. The changing ionospheric plasma parameter of electron density varies the resonant conditions of the probe.

$$V(f) = V_A cos(ft) \tag{1.5}$$

$$Z(f) = \frac{V(f)}{A} \tag{1.6}$$

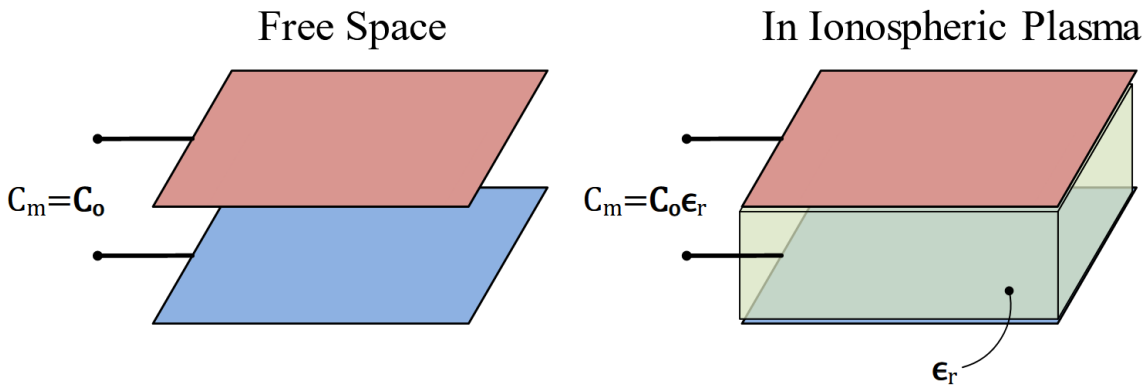

Fig. 1.4: Basic Concept of an Impedance Probe

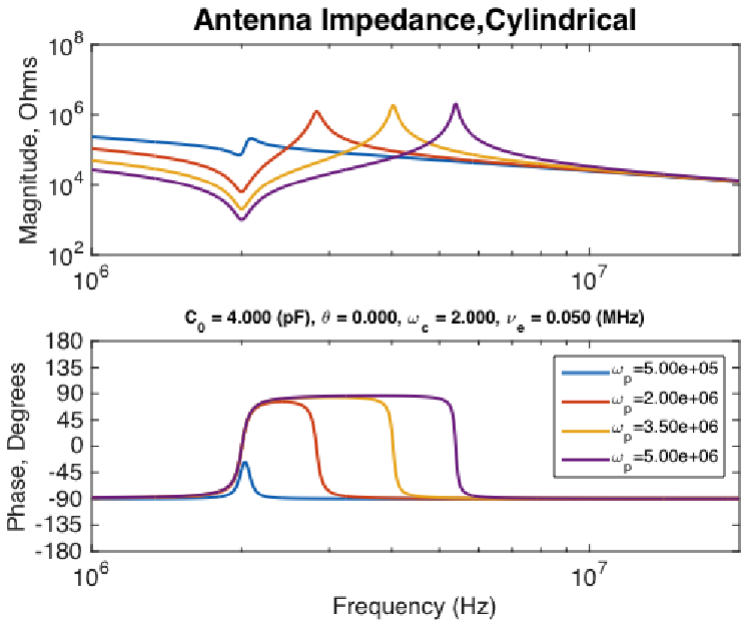

Fig. 1.5: Theoretical Impedance of Cylindrical Impedance Probe with Varying Plasma Densities [3]

The most common type of impedance probe geometry studied, both in theory and experimentally, has been a dipole or monopole which were electrically small relative to the free space electromagnetic wavelength at measurement frequency. To first order, the impedance as a function of frequency is only dependent on the average dielectric properties encompassed by the near field of the sensor. The driving voltage is kept small (10s of mV) to only perturb the surrounding electrons with energies that are slightly larger than their thermal potential. The system effectively behaves as if it were a capacitor, with ionospheric plasma as its insulating dielectric. The capacitance of the antenna is then the product of its value in vacuum, called the free space capacitance, C_0, and the dielectric effect of the plasma integrated over the surface of the capacitor. One approximation for the capacitance at a driving frequency $\omega = 2\pi f$, under a cold collisional plasma approximation for a cylindrical surface, is given by equation 1.7.

$$C = C_0 \frac{1}{2}\left(\left(1 + \frac{\omega_p^2(\omega - iv)}{\omega(\Omega^2 - (\omega - iv)^2)}\right)(1 + cos^2\theta) + \frac{\omega_p^2}{\omega(\omega - iv)} sin^2(\theta)\right) \tag{1.7}$$

In this expression, θ is the orientation to the magnetic field and Ω and ω_p are the well-known cyclotron and plasma frequencies. The current flowing through the capacitor shows two strong resonances, as illustrated in 1.5. One resonance is near the cyclotron frequency. The other resonance is at the upper hybrid frequency, ω_{uh}. Both have strong signatures in both magnitude and phase in their impedance vs frequency curves. The lower frequency has characteristics of a series R-L-C type resonance, and when driven disturbs the local plasma. Thus, it is not desirable to excite the cyclotron-sheath resonance for electron diagnostic purposes. The upper frequency has characteristics of a high-Q parallel R-L-C resonance and is used as the electron diagnostic feature. The local magnetic field strength determines the electron gyro frequency, Ω_e^2. The electron density is determined from the plasma frequency, $\omega_p^2 = w_{uh}^2 - \Omega_e^2$, where $\omega_p^2 = N_e e^2/m_e\epsilon_0$. The phase of the antenna impedance in dense plasmas, $N_e > 10^4$, changes abruptly at ω_{uh} from $-90°$ at frequencies above the resonance to $+90°$ below. The $0°$ phase point at the parallel resonance is tracked using a closed loop control system, resulting in absolute N_e measurements accurate to $\leq 1\%$. The IP will track the upper hybrid frequency, as well as points around the upper hybrid frequency to calculate the quality factor of the resonant peak.

1.2.3 Combining Langmuir Probe Measurements and Impedance Probe Measurements

Langmuir probes are useful for observing a relative change of electron density while a impedance probe can be used to determine the absolute electron density of ionospheric plasma. Figure 1.6 illustrates the problem by a comparison of three different measurement techniques from the FPMU instrument located on the ISS. Langmuir theory would suggest that the absolute electron density can be computed from the DC current of the probe. Unfortunately there are multiple limiting factors that are not accounted for by theory yet are experimentally known to exist and to be important [19–21]. Primary among them is the ion wake, which forms behind objects moving at orbital velocities. This effect can modify the collecting area, A, by an attenuation factor of 0.5. TThe orientation of the probe with respect to the magnetic field and the probe's work function (i.e., surface contamination)

are significant measurement factors. Other important effects include spacecraft charging, photoemission currents, energetic particle impacts, as well as extraneous exposed potentials on the spacecraft surface from exposed connectors or solar cell interconnects [22–27]. The absolute N_e calibration of fixed bias Langmuir measurements can be accomplished by cross calibrating against the impedance technique for electron density measurements. The electron temperature measurements can be corrected with simultaneous floating probe measurements to monitor change of spacecraft charging during the voltage sweep. Examples of this calibration for past flights are shown in Figure 1.6 and Figure 1.7.

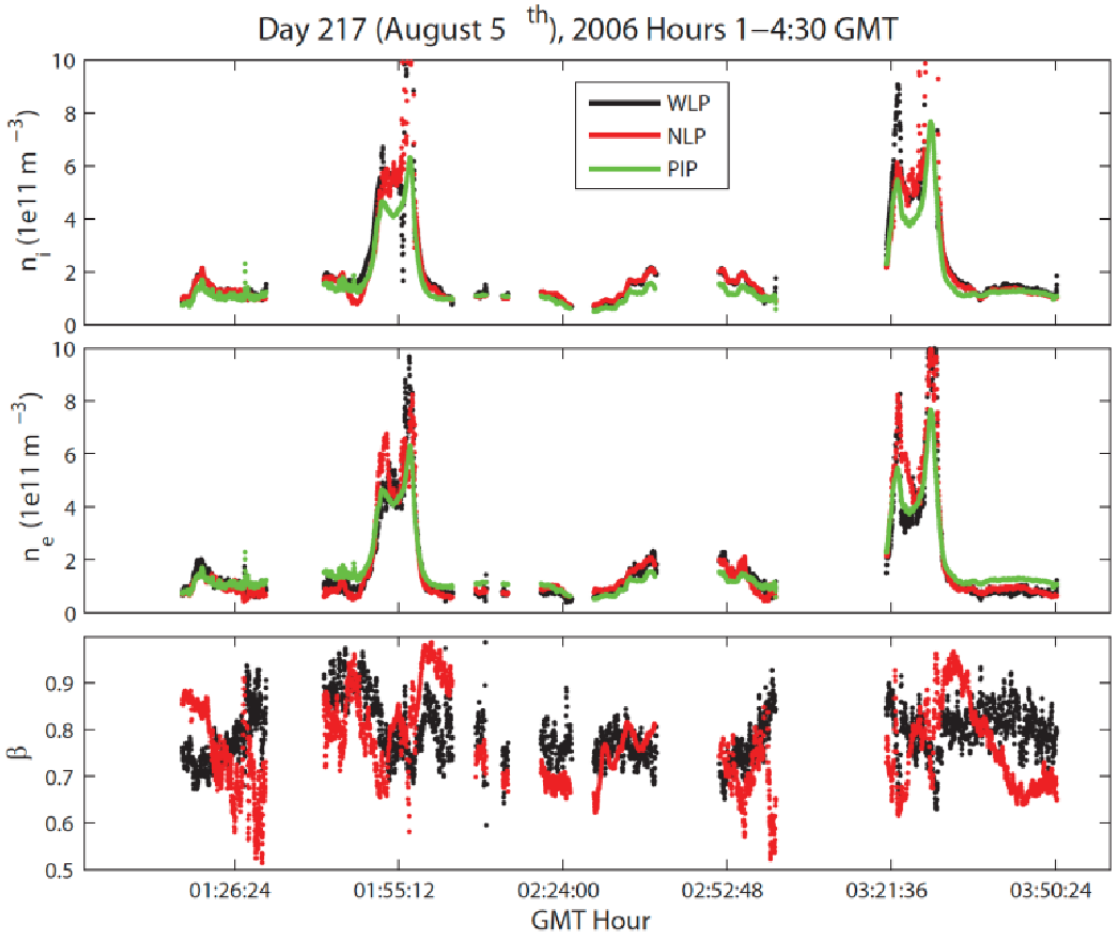

Fig. 1.6: A Comparison of Electron Density Measurements Derived from two different Langmuir measurements and an Impedance Probe from the FPMU instrument on the ISS [4]

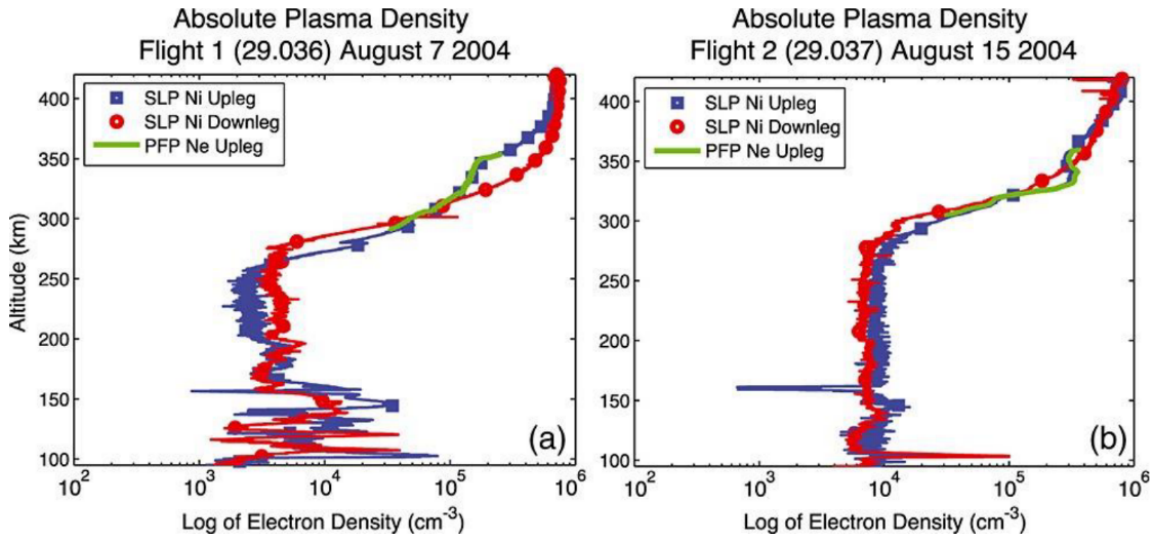

Fig. 1.7: The tracking RF impedance technique used to calibrate the sweeping Langmuir measurements which are then extended into low electron density regions at low altitude [5]

1.3 Floating Potential Probe and Electric field Wave Spectrometer

The FPP and EFW use the same physical probes. These probes act as a voltmeter that measures the potential of an isolated conductor, the probe surface, relative to the spacecraft body. Both the spacecraft and the isolated conductor arrive at a charge equilibrium with the space environment such that the currents due to thermal ions and electrons are balanced with photoelectrons and high energy particles. This is the potential in Langmuir theory where the net collection current of ions and electrons is zero to the probe surface. They each float to a few volts negative in the ionosphere, depending on the plasma temperature, with some differences due to size, shape, illumination, and surface properties. For the purposes of SWP2 we are only interested in observing changes in the spacecraft potential relative to the environment that occur during the brief period of a Langmuir sweep, thus the potential of the isolated conductor is a constant reference point during the sweep.

The local electric field between two probes is measured using equation 1.8 where V is the voltage difference between the two probes and d is the distance between the probes.

$$E = \frac{V}{d} \tag{1.8}$$

1.4 Future Missions for Space Weather Probes 2

SWP2 is the successor of the SWP instrument flown on SPORT. The success of SWP as a compact and integrated electronics package for observing ionospheric plasma has made an improved version of the probe desirable. The original SWP contained two FPPs used to monitor both spacecraft charging and one component, or direction, of the electric fields when used as a pair. The SWP2 contains four FPP measurements that can be configured to measure three orthogonal components of the electric field on a CubeSat. The SWP contained a single Langmuir probe that was time multiplexed between DC electron saturation current measurements and IV sweep measurements. It also had non-linearities in the probe circuit at high current levels and a high noise floor in the high-gain measurement channel. SWP contains two separate Langmuir probes that are to be used independently for measurements of the electron saturation current and for creating IV curves and corrects the noise problems. There are improvements to the impedance probe as well as the on-board magnetometer and the overall power consumption of the instrument suite.

SWP is planned to fly on several upcoming and proposed CubeSat missions. The ACMES mission, planned to launch in 2024, features the Planar Langmuir And Impedance Diagnostic (PLAID) probe which is a version of SWP2 connected to planar surfaces for the probes instead of deployed booms and cylindrical sensors. This will fly on the three axis stabilized ACMES spacecraft being mounted on the ram face. ITASAT-2 is a proposed as a follow-on mission to SPORT. This joint mission between the United States of America and Brazil will use the SWP2 instrument on each of three satellites flying in a constellation. OUIJI is a mission proposed to DARPA that will feature the SWP2 package on a constellation of up to 5 spacecraft flying below the peak density of the ionosphere.

CHAPTER 2

Literature Review

2.1 Signal processing hardware design in Simulink

A number of authors have looked at using various high-level tools to develop digital signal processing chains and system controllers for FPGAs. There are a variety of these higher level tools chains now in existence which compile or create synthesizable code such as VHDL or Verilog as an intermediate step to in programming a FPGA. Notables are the various efforts to translate the C programming language to a hardware description language such as Bach-C (Sharp), C2H (Altera), C2Verilog (Syn- posys), Catapult-C (Mentor Graphics), etc. MathWorks has developed their HDL Coder product to translate the MATLAB and Simulink language and system description into either synthesizable VHDL or Verilog. Users can use pre-built library blocks optimized for use on FGPAs as well as design custom blocks and state machines in Simulink's state flow environment to build a digital design. Simulink has been used for compiling DSP systems into VHDL since 1999 [28]. Kurowski et. al. designed a polyphase IIR lowpass filter in Simulink and compared the design to a handwritten VHDL design. They noted several shortcomings in HDL coder which have since been overcome in more recent versions of HDL Coder. Sarge evaluated Simulink HDL coder as an alternative workflow for producing hardware description [29]. They found that the generated hardware does not perform quite as well as a hand-optimized design, but that it does perform well enough to be practical and can be capable of greater flexibility in structure than a design created with a more traditional workflow. Pongratz and Cherian found that in big designs HDL Coder did not have a problem meeting the timing requirements and the results were comparable with hand coding in VHDL [30]. However, they found that the VHDL produced by HDL Coder does not necessarily map optimally to the resources on the FPGA fabric and it is difficult to change the mapping of resources according to constraints.

They also found that HDL coder was faster and easier to learn, work with and provided many optimization methods can are done automatically, and that its use may increase productivity for development teams. Caseiro et al used MATLAB/Simulink to develop power electronics systems based around Zynq FPGA development boards [31].They found that this methodology considerably reduced the development time and effort, but also found some important drawbacks and hurdles in Simulink/HDL Coder development. They found both solutions and design techniques that overcame these drawbacks and allowed for the implementation of all required peripherals (ADCs, digital outputs, etc.), system protections, and real-time data acquisition. Simulink and HDL coder was also used by USU to develop a part of the firmware for the impedance probe sensor on the SPORT mission [32]. They found that the libraries of complex mathematical operations and signal processing enabled the realization of a design in a much shorter period that would be possible with hand coding in HDL. They also found that Simulink provided a level of self-documentation and provided for the potential of reusable code and more maintainable code in the university setting.

CHAPTER 3

Theory of Operation

This chapter outlines the requirements for each probe as well as the theory behind how the hardware design operates.

3.1 Measurement Requirements

The sampling requirements for each probe data outputs are shown in table 3.1. The digital design must be constructed to make raw measurements and process the signals to provide data samples at the rates specified. The measurement ranges of voltages and currents as well as the noise levels and measurement precision are set by the analog design, which is not covered in this thesis. Essentially this thesis starts with the data produced at the analog to digital converter or the signals presented to digital to analog converters and the control of the signals from each of the probes in SWP2.

Probe	Parameter	Sample Rate
FPP	Voltage Measurement	100 Hz
EFW	Voltage Spectrogram	10 Hz
FLP	Current Measurement	100 Hz
	Current Spectrogram	10 Hz
SLP	Voltage Sweep	1 kHz
	Current Measurement	10 kHz
	Langmuir Sweep	1 Hz
IP	Frequency Measurements	100 Hz

Table 3.1: Measurement Requirement Table

3.2 System Architecture

The SWP2 architecture consists of parallel signal processing chains for each probe as illustrated in figure 3.1. The processing chain for each probe can be divided into ADC or DAC drivers, signal processing blocks, and packetizing blocks. The raw voltage data from the sensors analog electronics chain is read in by an ADCs. If the sensor has a voltage or signal output as the SIP and SLP do, then the output voltage signals are driven via DACs. The raw digital data is then processed and formed into packets of data, using the CCSDS space packet protocol for telemetry to the ground. Some probes produce multiple data packet types, the full description of each of these telemetry packet types is documented in an Excel telemetry dictionary file and an extract from that file is presented in appendix appendix A. Telemetry packets are produced in parallel and presented asynchronously to a routing function within the FPGA. Packets are stored in a circular buffer and then transferred to the spacecraft upon demand. If the spacecraft does not request packets at a sufficient rate, then the circular buffer will overwrite itself and data will be lost. The processing chains for the probe have various configurable settings and parameters can be changed while the probes are in operation without re-imaging the FPGA. The setting of the SWP2 system parameters is controlled by receiving CCSDS space data packets from the spacecraft that contain parameter values to override the default values contained in the firmware. This chapter focuses on the design requirements of the control and processing for the FPP, EFW, SLP, FLP, and IP probes.

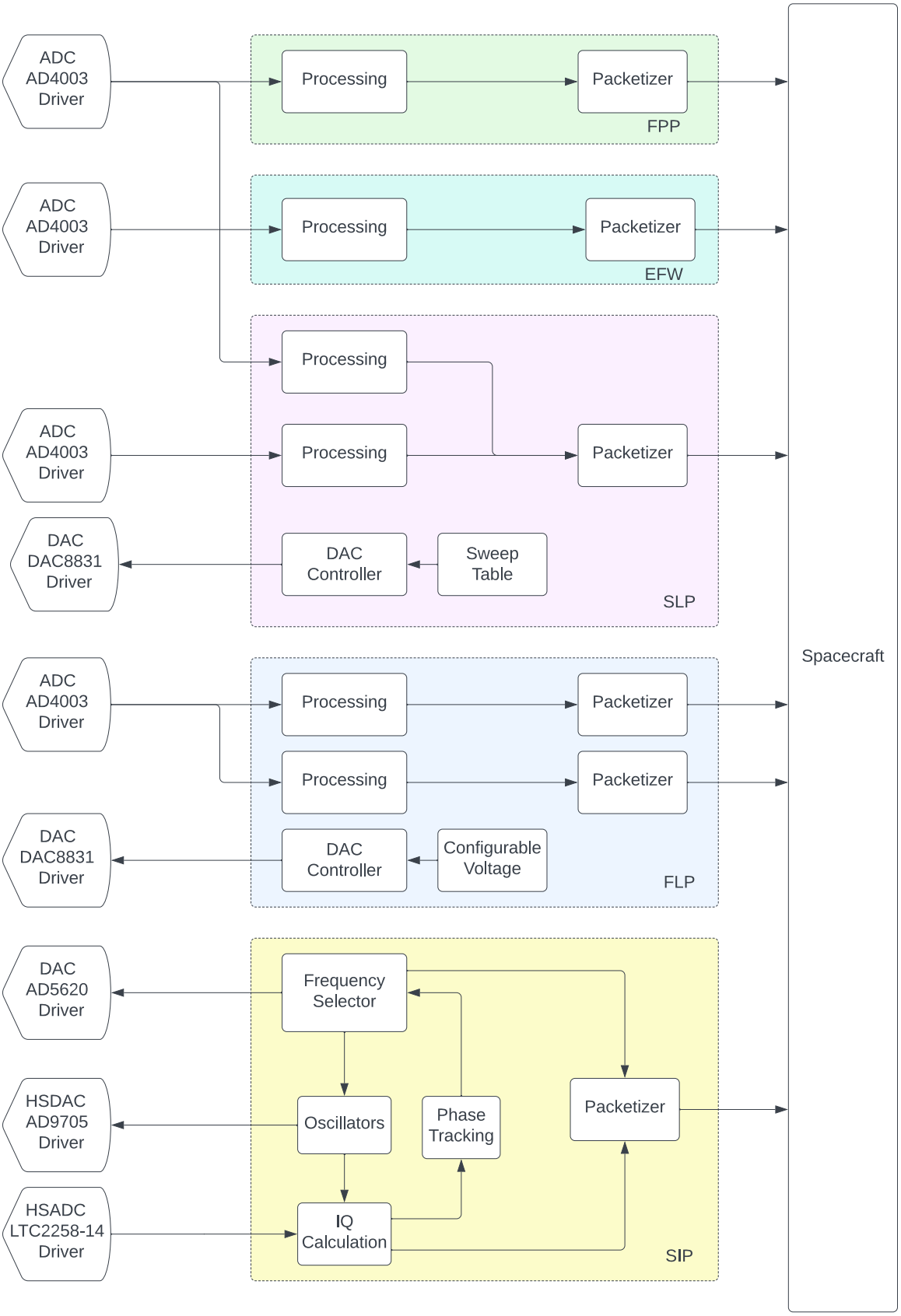

Fig. 3.1: SWP2 Firmware Overview

3.3 Floating Potential Probe (FPP)

The floating potential probe is a set of four probes that measure the potential between the probe surface and the spacecraft. The probes are placed to provide measurement baselines in three orthogonal directions which is accomplished by deploying booms that are orthogonal to the sides of the satellite at both the ram and trailing faces of the spacecraft.

The FPP uses four copies of the same processing chain, one for each probe, that produce two almost identical packets, one packet containing data from probes 1 and 2, and another packet for containing data from probes 3 and 4. The high level design for the FPP processing chain is shown in figure 3.2. The design takes in raw voltage data as an 18 bit values from the ADC at a 200kHz rate from each of the four simultaneously sampled probes. The data is then averaged and down sampled to a rate of 100 Hz to reduce noise in the measurements. The resulting data is output as a 20-bit signed value. The average and down sample operation performs a lowpass filter operation on the data, getting a DC measurement of the potential between the probe and the spacecraft. The down-sampled data is then sent into FPP probe packets. DC electric field measurements are made by differencing the data from opposing floating potential probes.

Fig. 3.2: FPP High Level Design

3.4 Electric Field Wave Spectrometer (EFW)

The electric field wave spectrometer operates on a signal that has been differenced and high pass filtered by the analog signal chain prior to the ADC. There are two separate spectrometers, one for sensors 1 and 2 and the second for sensors 3 and 4. The high pass cutoff is nominally 30 Hz thus removing the larger quasi-DC signal from the smaller signal of electric field waves. The filtered and gained difference signal is then passed into the digital design by an ADC. The high-level design for the EFW is shown in figure 3.3. The potential difference between the probes is sampled at a 200 kHz rate as 18-bit numbers.

That data is then sent into an FFT of size 2048 to create 1024 unique frequency bins of the signal amplitude. The EFW calculates the power spectral density (PSD) of the signal taking the magnitude squared of the positive frequency bins and doubling the value as seen in equation 3.1 where $F^+(f)$ is the positive frequency bins from the FFT. The 1024 electric field wave power bins are then summed into 16 bands by adding the power of adjacent bins together to form the total power in each of 16 different frequency bands. The resulting spectral bands are output as a length 16 vector of 24-bit numbers.

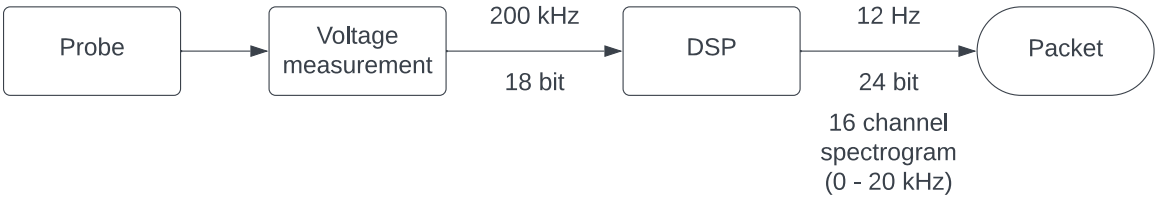

Fig. 3.3: EFW High Level Design

$$PSD(f) = 2|F^+(f)|^2 \tag{3.1}$$

3.5 Fixed Bias Langmuir Probe (FLP)

The Fixed Bias Langmuir Probe is biased positive to collect electrons with the probe output voltage being held at a fixed level. The voltage is chosen to be in the electron saturation region of the Langmuir IV curve to provide a relative electron density signal. The high level block diagram for the FLP is shown in figure 3.4. The FLP produces two types of data packets, the fixed bias probe packet and the fixed bias wave spectrometer packet. Between these two packets the probe provides information on electron density and density fluctuations from DC to 20 kHz.

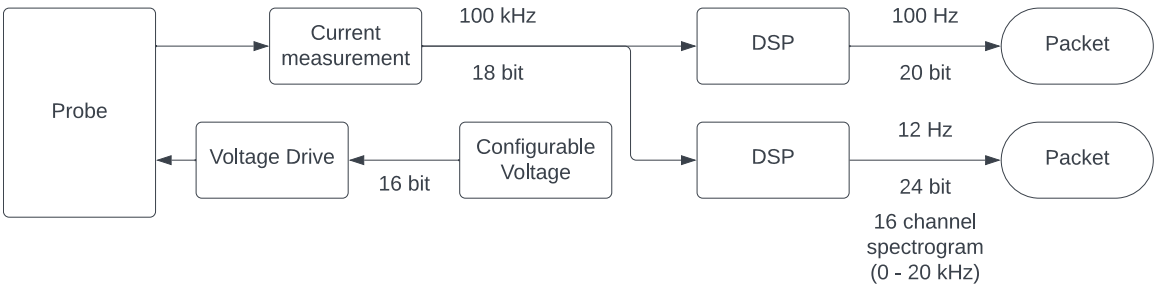

Fig. 3.4: FLP High Level Design

3.5.1 Fixed Bias Langmuir Probe

The DC to 50 Hz packet produced is called the Fixed Bias Probe packet. For this packet, the current is sampled at a 100kHz rate as 18 bit signed values from the ADC. This data is then averaged and down-sampled to a rate of 100 Hz to reduce noise in the measurements and to increase the dynamic range of the current measurement. The down-sampled data is then formatted into FLP probe packets as 20 bit signed numbers.

3.5.2 Fixed Bias Langmuir Wave Spectrometer

The other packet produced by the fixed bias probe is called the fixed bias wave spectrometer (FBW). The FBW provides the spectral properties of the current measurements from the fixed bias probe. The digital design for the FBW is identical to the design for the EFW, except the data is read at a 100kHz rate from the Fixed Bias probe. The current measurements are 18 bit signed numbers. The FBW produces the power spectral density of the Langmuir probe current fluctuations in 16 different frequency bands as 24 bit numbers.

3.6 Sweeping Langmuir Probe (SLP)

The Sweeping Langmuir Probe adjusts the voltage of the probe in steps and measures the resulting current at each step to produce an IV curve. Simultaneous with the current measurement the floating potential probes FPP-1 and FPP-2 are sampled to monitor the potential of the spacecraft. Both sets of data are recorded in the SLP packet. The high-level design for the SLP is shown in figure 3.5. The SLP steps the voltage using a 16-bit DAC at a

2 kHz rate or 500 us per step. The current collected by the surface of the probe is sampled at a 200 kHz rate as a signed 18-bit value simultaneously with the FPP voltages. The current, and voltage, samples are then filtered to remove the transients noise and increase the dynamic range at each step. The resulting filtered current samples are signed 20-bit values placed into the data packets. The SLP also controls the gain of the analog amplifier circuit by using a boolean to select between two gain values. The low gain is optimized for the electron portion while the high gain setting is optimized for the ion portion of the IV curve.

The SLP has different operational modes that produce different packets. The different modes can be seen in table 3.2. In science mode, the SLP triggers two different types of sweeps, fast sweeps and full sweeps, which each create a different packet. The full sweep sweeps over the full voltage range set for the IV curve measurement. The fast sweep is focused on taking a minimum number of samples in the key ion saturation, retarding, and electron saturations regions so that electron temperature can be obtained at a high cadence and low data rate. The fast sweep takes 2 samples in the ion saturation region, 2 samples in the electron saturation region, and sweeps over the electron retardation region to measure these key parameters of the IV curve with much fewer data points than the full sweep. The timing for science mode is shown in figure 3.6.

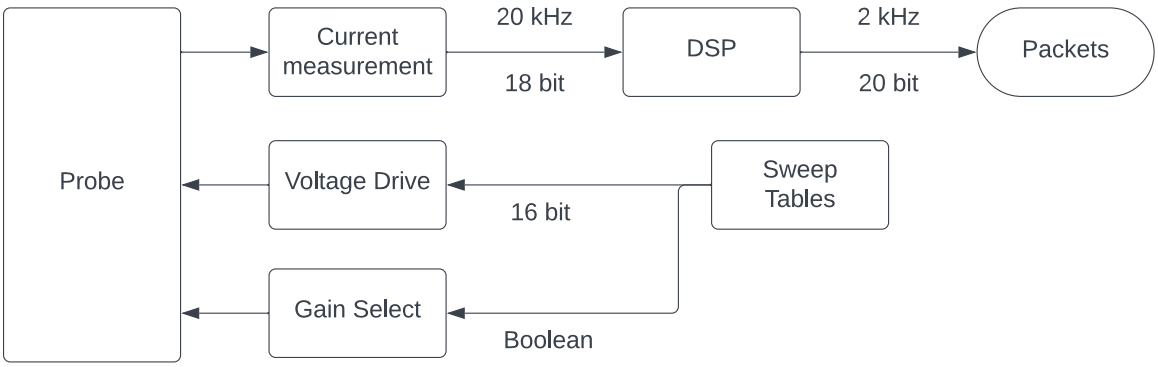

Fig. 3.5: SLP High Level Design

Mode	packets produced
Science Mode	FULL packet, FAST Packet
Calibration Mode	FULL packet

Table 3.2: SLP Modes and Packets Generated

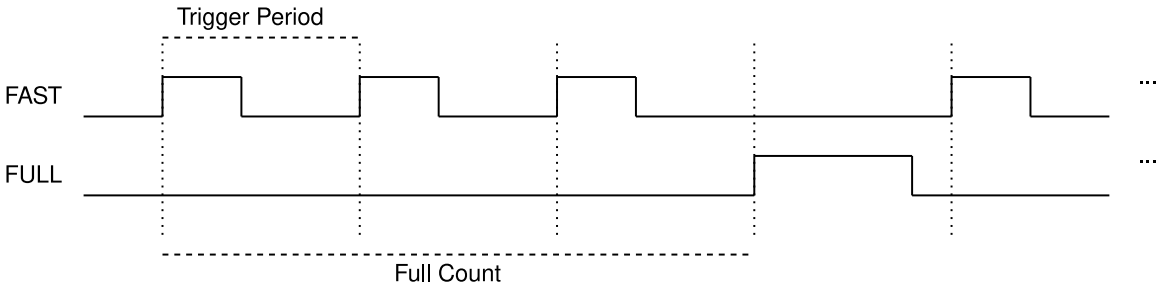

Fig. 3.6: SLP Science Mode Timing

3.6.1 Full Sweep

The full sweeping mode sweeps the probe over the full range of voltages supported. A sweep starts at 3V and sweeps downwards to -2V with 0.01mV steps. The sweep then sweeps upwards from -2V to 3V at the same rate. The sweep then holds the voltage at 3V for 50 steps, as seen in figure 3.7. At each step, a gain select value is specified as well. At each voltage step, 10 current samples are combined and processed to make one filtered current sample. The current at each voltage step is sent to the SLP full packet. To calibrate the SLP current samples, at each voltage step, an FPP measurement is made and added to the SLP Full packet.

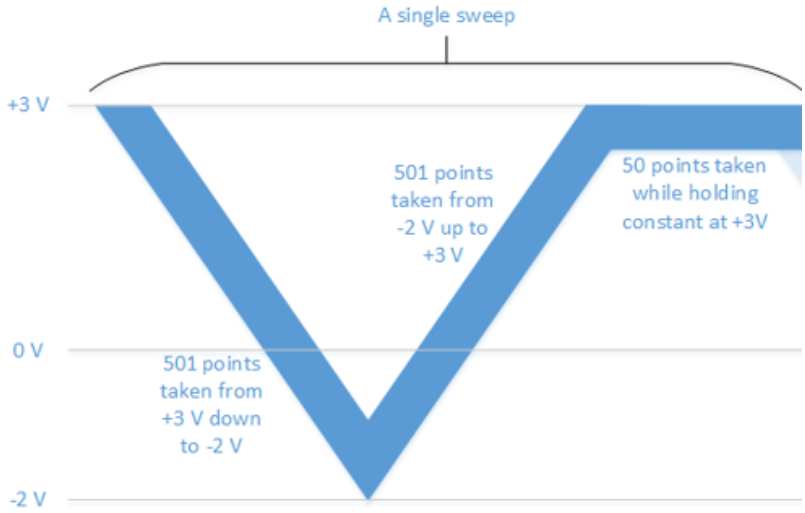

Fig. 3.7: SLP Full voltage Sweep

3.6.2 Fast Sweep

The fast sweeping mode first sweeps the probe over a small range of voltages at the extremes of the voltage range, namely $-2V$, $-1.99V$, $2.99V$, and $3V$. The probe then sweeps over 28 voltage offsets from the floating potential of the Langmuir probe. The fast sweep points are shown on a Langmuir IV curve seen in figure 3.8. At each step a gain select value is specified. The floating potential of the Langmuir probe is estimated by taking a sample of the floating potential from the FPP and adding a calibrated offset to the value.

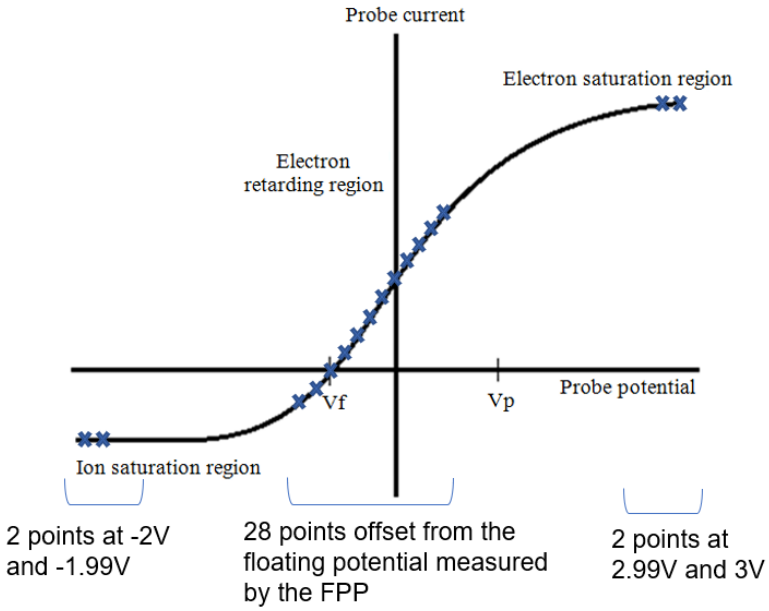

Fig. 3.8: SLP Fast voltage Sweep

3.6.3 Calibration Mode

In Calibration mode, the SLP outputs full sweep packets at a configurable rate. The probe normally sweeps through all steps of the full table, but it can be configured to stay at a fixed table index or fixed voltage using a telecommand packet. This mode is used to verify the output voltage on the probe at each step number of the sweep. It is also used to verify the gain and linearity of the current measurements.

3.7 Impedance Probe (IP)

The IP measures the impedance of a probe immersed in the ionospheric plasma as a function of frequency in the 0.1 to 30 MHz range. The data consists of in-phase, I-samples, and quadrature, Q-samples, of the current to the probe at a set frequency steps when driven with a constant voltage signal. The probe also tracks the parallel resonant condition of the probe and outputs the frequency at which the resonant condition occurs. To measure the Q-factor of the resonant peak, the IP also produces a hybrid packet that includes the frequency of the parallel resonant peak and a set of impedance values around the resonant

peak. The high-level design for the IP is shown in figure 3.9. The digital design for the IP feeds a 16-bit high speed DAC that drives the RF voltage at the probe. The digital design also controls a second 16-bit DAC that sets the amplitude of the voltage signal applied to the probe. The IP takes in the measured current from the probe surface through a 14-bit high speed ADC. The digital design reads the input current waveform from the high-speed ADC and processes this waveform to produce the vector current measurements representing the impedance of the probe.

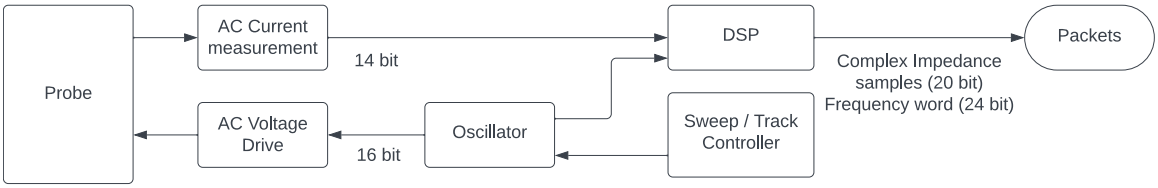

Fig. 3.9: IP High Level Design

The IP has different operational modes that produce different data packets as seen in table 3.3. The four submodes of the IP and the four packets produced are short sweep impedance probe (SIP), the long sweep impedance probe (LIP), the tracking impedance probe (TIP), and the track-Q impedance probe (QIP). The SIP sweeps over a range of fre- quencies that includes the upper parallel resonant condition, but not the lower series resonant condition of the impedance curve. When the impedance probe drives the series resonant condition, the probe energizes and heats the plasma, potentially resulting in measurement errors in the other probes due to this disturbance. Therefore the lowest frequency of the IP is set to avoid exciting at the lower series resonance condition. However, occasionally the IP will do a longer sweep that includes the lower resonant condition to study the physics of this resonance and for comparison with theory. This longer-range sweep that include the lower frequencies is called the LIP. The TIP tracks the frequency of the parallel resonant condition of the probe which is associated with the upper-hybrid frequency of the plasma as a 24-bit value of the frequency. The QIP is a hybrid packet that adds additional information to the tracking of the parallel resonant frequency. Once the parallel resonant

frequency is determined, the QIP steps around the resonant frequency and measures the current around the resonant peak. This allows for a determination of the Q-factor of the upper resonant condition in a small data packet an at a higher precision and rate than what can be done with the SIP packets.

Mode		Acronym	Packets Produced
Science Mode	Tracking Mode	TIP	TIP packet
	Tracking Q Factor Mode	QIP	QIP packet
	Short Sweep Mode	SIP	SIP packet
	Long Sweep Mode	LIP	LIP packet
Calibration Mode		CIP	SIP packet

Table 3.3: IP Modes and Packets

3.7.1 Science Mode of the IP

When the IP instrument is in science mode, the instrument switches between different sub modes as seen in figure 3.10. The instrument switches between a tracking section and a sweeping section. In the tracking section, the instrument can be in either the TIP submode, or the QIP submode. In the sweeping section, the instrument will either be off, in the SIP submode, or in the LIP submode. The timing for when the instrument switches modes is shown in figure 3.11. The trigger period is the period between two track sections. The Q variable is how many TIP track sections are between QIP track sections. The S variable is how many sweep sections with the instrument off are between SIP sweep sections. The L variable is how many SIP sweep sections are between a LIP sweep section.

Fig. 3.10: IP science mode submodes

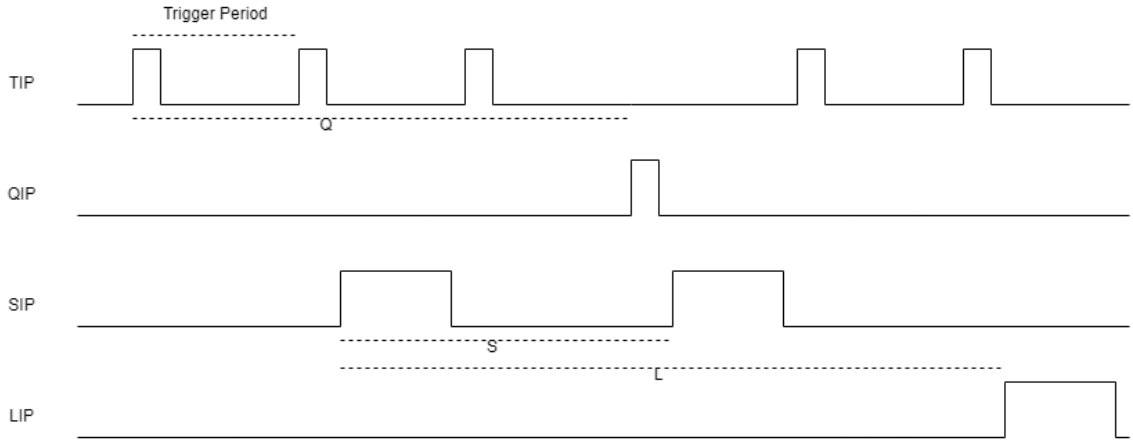

Fig. 3.11: IP Science mode timing

SIP mode

The SIP sweeps the probe over the range of frequencies from $2MHz$ to $30MHz$ in 512 logarithmically spaced frequency steps. The output voltage is given by equation 3.2 where $f(t)$ is the step frequency. The SIP sweep is designed to sweep through the upper parallel resonant condition without sweeping through the lower series resonant condition.

$$v(t) = Acos(2\pi f(t)t) \tag{3.2}$$

At each frequency step, the digital design inputs the measured current waveform. The impedance at each frequency step is calculated from in-phase (I) and quadrature-phase (Q) values as seen in equation 3.3 where j is the imaginary unit. In-phase and quadrature-phase calculation is a common signal processing technique in RF applications. The input waveform is multiplied by $Acos(2\pi f(t)t)$ and lowpass filtered to get the I value, and multiplied by $-Asin(2\pi f(t)t)$ and lowpass filtered to get the Q value. The I and Q values for each frequency step are put into the SIP packet as 20 bit words.

$$Z = I + j * Q \tag{3.3}$$

LIP mode

The LIP sweeps the probe over the range of frequencies from $0.5MHz$ to $30MHz$ in 600 logarithmically spaced frequency steps. The LIP sweep is the same as the SIP sweep with 88 frequency steps added to the beginning of the sweep. The impedance at each frequency step is measured using the same method as the SIP, outputting the I and Q values as 20 bit words.

Tracking Loop

The IP will track the upper resonant condition using a phase tracking loop. The tracking loop is used in the TIP and QIP submodes. The high-level design for the tracking loop is shown in figure 3.12. The probe voltage is controlled by an oscillator driven by a frequency word. The phase of the probe impedance is calculated from the I and Q values and fed into a loop filter. The output of the loop filter is sent to an accumulator which controls the frequency word.

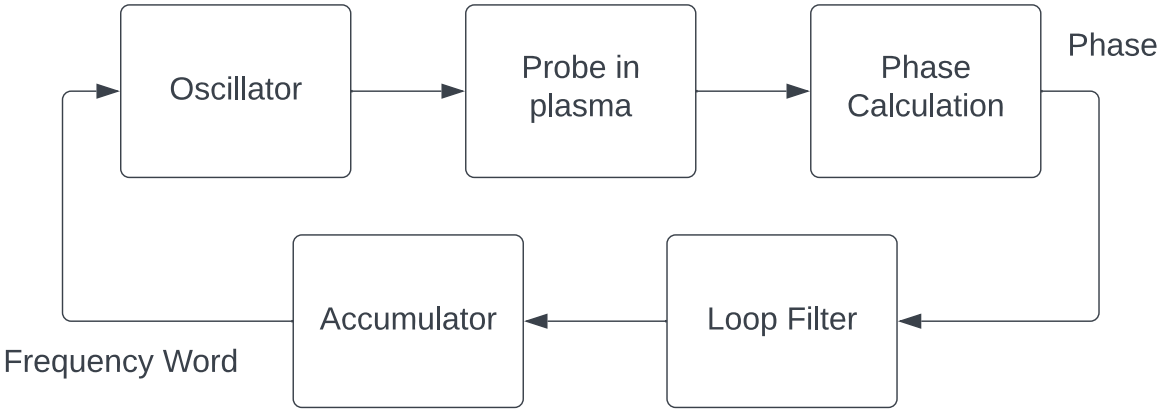

Fig. 3.12: Tracking Loop

TIP mode

The TIP tracks the upper resonant condition, the frequency at which the probe has a high impedance resonance and the phase of the impedance transitions from inductive to capacitive through zero, using the tracking loop. The TIP does this by driving the output

waveform at an initial frequency. The tracking loop is then enabled and controls the output frequency. After the frequency has settled, the resonant frequency is put into the TIP packet as a 24-bit word representing the frequency.

QIP mode

The QIP mode is a hybrid of the TIP mode and the sweeping modes. In QIP mode, the upper resonant condition is tracked using the tracking loop. The probe is then swept over a table of frequency offsets from the upper resonant frequency. The I and Q values are measured at each frequency offset. The frequency is sent to the QIP packet as a 24-bit value, and the I and Q values at each frequency offset are sent to the QIP packet as 20-bit values.

3.7.2 Calibration mode

The calibration mode produces IP packets that are useful to calibrate the probe. The calibration is commanded by sending CCSDS space packets to the IP to switch between sweeping through the SIP frequency table and setting the frequency to a specified frequency value. This allows a computer to configure the impedance probe and then record the I and Q output with different test sources connected to the probe. This is done to determine the transfer function of the probe without the presence of plasma called the free space impedance. The uncalibrated amplitude and phase of the driving RF signal at each frequency is measured and then saved in a calibration table withing the instrument so that the output of the probe is driven with a constant amplitude and phase at all frequencies.

CHAPTER 4

Design

This chapter discusses the processing design for each of the probes.

4.1 Generic Design

The generic design for a processing chain is shown in figure 4.1. Each packet's processing chain starts with drivers that get the raw data from the sensors. The raw data is then sent to processing blocks that process the data. The processed data is then converted into data granules and into packets as shown in appendix A. A data granule is a single sample of processed data packed into a byte structure. Data granules are compiled into packets. The packets are sent through a packet router and then sent to the spacecraft.

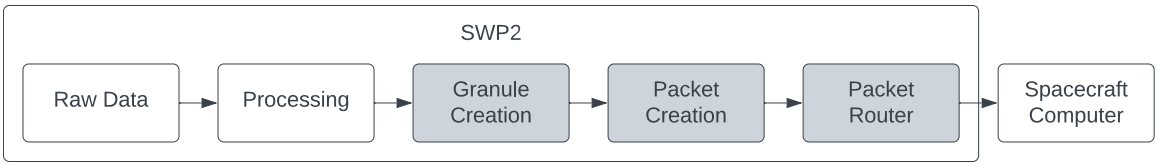

Fig. 4.1: Generic Data Processing Chain

Each probe has configurable parameters that control the operation of the probe. Some of these parameters are tele-command configurable meaning that the operating parameters can be changed from their default values without reconfiguring or reprogramming the FPGA.

4.2 Floating Potential Probe (FPP)

The FPP processing uses an average and down-sample filter to remove noise from the data. The Simulink design of the average dawn sample block is shown in figure 4.2. The average and down-sample block operates by accumulating a specified number of values read

from the FPP ADC into one value. The accumulated value is then shifted and sliced into a 20-bit container. The 20-bit values from the different FPP probes are assembled into a data granule. The FPP does this by using a discrete time integrator block in Simulink. When the data is valid, the integrator operates on the data, otherwise the integrator is given a 0. The output of the integrator is shifted and sliced to form the output data. To control the timing of the block, a counter tracks the number of input values. When the count reaches the down sample factor, the output is marked as valid, and the counter and the integrator are reset.

The processing is controlled by configurable parameters as seen in table 4.1. The parameter FPP.ADC_Sample_Period controls the rate at which raw data samples are taken from the ADC over SPI. The parameter FPP.Accumulate_Count controls how many raw data samples are accumulated into one granule. The parameter FPP.Digital_Gain controls which bits from the accumulated value are sliced and put into the FPP data granule.

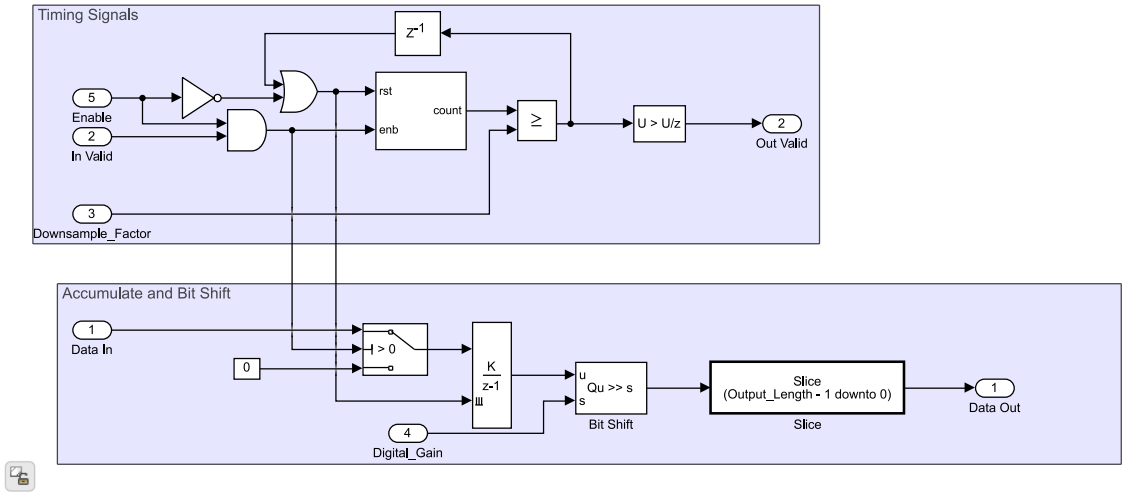

Fig. 4.2: FPP Average Downsample

Parameter Name	Data Type	TC*	Description
FPP.ADC_Sample_Period	uint32	n	Number of FPGA ticks between FPP ADC sample triggers
FPP.Accumulate_Count	uint12	n	Number of FPP samples to accumulate
FPP.Digital_Gain	uint8	y	2^N gain of the accumulated value

*Tele-command Configurable

Table 4.1: FPP Configuration Values

4.3 Electric Field Wave Spectrometer (EFW)

The EFW creates a spectrogram of the voltage between two floating potential probes. The EFW processing block takes in 2048 raw samples and performs a 2048 FFT. The power spectral density is calculated by taking the magnitude squared of each positive frequency bin. To combine the positive and negative frequencies, we then take only the positive frequencies and bit shift to double the value. The bins are then combined by summing the values of the frequency bins into 16 frequency bands. The 16 bands are sent into a granule. The processing for the EFW is shown in figure 4.3. The processing uses a simulink 2048 FFT block to perform the FFT. The 1024 positive frequency bins are magnitude squared and doubled to form the PSD. The bins are fed into a FIFO that the summing state machine uses. The EFW uses two length 16 vectors that control how the FFT bins are combined into bands, the start bin vector and the end bin vector. The i^{th} output frequency band is the sum of all PSD bins from start bin vector[i] to out bin vector[i].

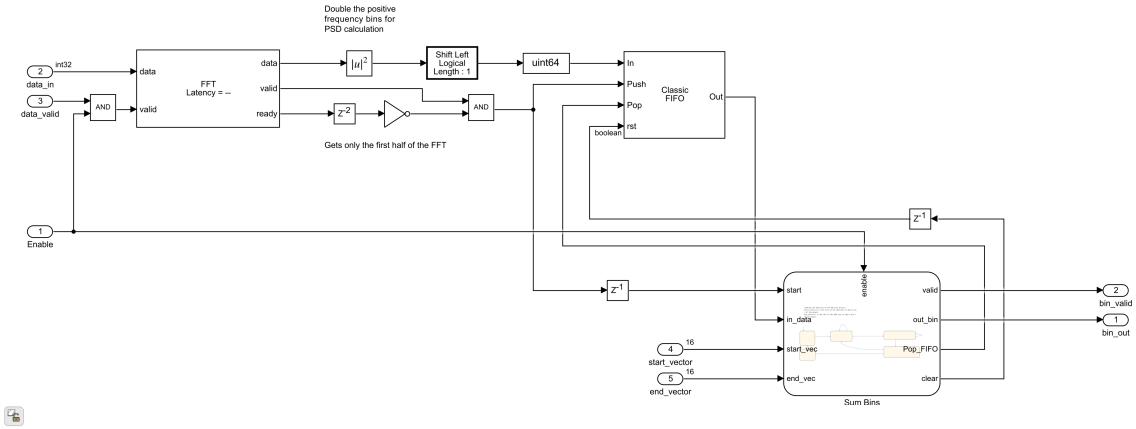

Fig. 4.3: EFW Processing

The state machine used for combining the frequency bins is shown in figure 4.4. The state machine uses two indices, the input bin index, m, and the output band index, n, to keep track of the summing. The input bin index tracks the 1024 input bins that are being summed. The output band index tracks the 16 output bands that the input bins are being summed into. When the state machine is enabled and a start signal is sent, the state machine enters a start sum state. In this state, the state machine increments m and pops the FIFO until m is equal to the n^{th} value of the start bin vector. The state machine then enters the summing state. In this state the state machine keeps popping values from the FIFO and sums the values until m is the n^{th} value in the end bin vector. The state machine then enters the output sum state. In this state, the state machine stops popping from the FIFO, outputs the summed value and increments n. If n index is greater than 16, all 16 output bins have been summed, the FIFO is emptied, and the state machine waits for another start signal. Otherwise, the state machine returns to the start sum state

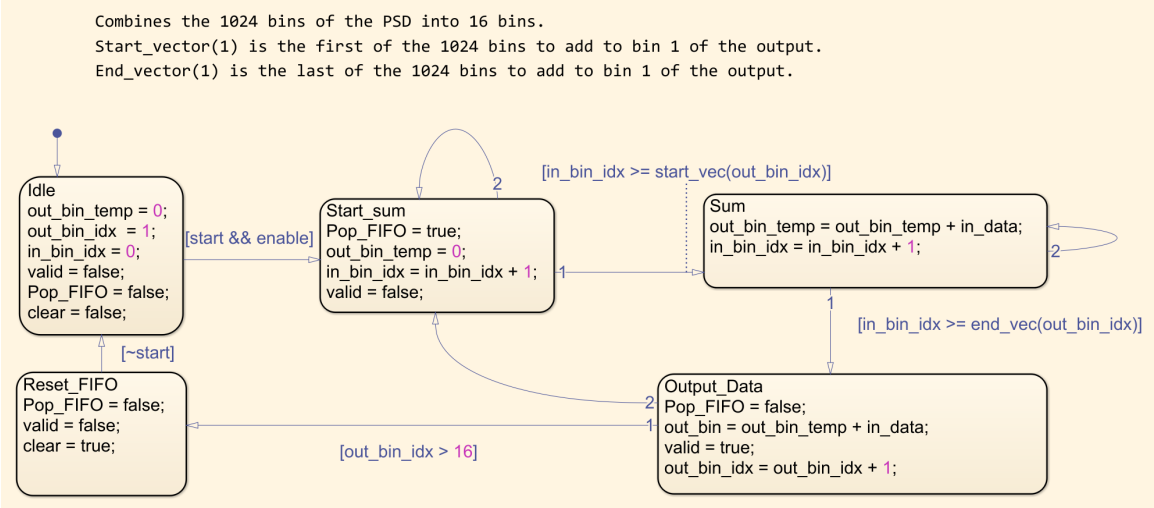

Fig. 4.4: EFW Summing State Machine

The operation of the EFW is controlled by parameters listed in table 4.2. EFW.ADC_Sample_Period controls the rate at which raw data samples are taken from the ADC over SPI. EFW.Start_Bins and EFW.End_Bins are the start bin vector and the end bin vector.

Parameter Name	Data Type	TC*	Description
EFW.ADC_Sample_Period	uint32	n	Number of FPGA ticks between EFW ADC sample triggers
EFW.Start_Bins	uint16 [16]	n	Vector of the starting FFT bins for each of the 16 spectrometer bins
EFW.End_Bins	uint16 [16]	n	Vector of the ending FFT bins for each of the 16 spectrometer bins

*Tele-command Configurable

Table 4.2: EFW Configuration Values

4.4 Fixed Bias Langmuir Probe (FLP)

The FLP has two parallel processing paths that produce two different packets: the fixed bias probe (FLP) packet which contains DC measurements of the fixed bias Langmuir

probe, and the fixed bias Langmuir wave spectrometer (FLW) packet which contains AC measurements of the fixed bias Langmuir probe.

4.4.1 Fixed Bias Probe

The FLP packet processing uses an average down-sample filter to remove noise and make DC measurements. The average down-sample filter used for the FLP packet is the same as the average down-sample filter from the FPP processing seen in figure 4.2. The FLP is controlled by configurable parameters seen in table 4.3. FLP.ADC_Sample_Period controls the sample rate of the FLP probe. FLP.Accumulate_Count and FLP.Digital_Gain control the number of samples to accumulate and the number of digits to bit shift the accumulated result. FLP.DAC_Word controls what voltage the FLP surface is held at.

4.4.2 Fixed Bias Langmuir Wave Spectrometer

The FLW packet processing takes a 2048 FFT of the raw FLP data. The processing then calculates the power spectral density of the FLP data. The PSD bins are then combined into 16 bands. The FLW packet processing has the same design as the EFW processing as seen in figures 4.3 and 4.4. The FLW is controlled by configurable parameters seen in table 4.3. FLW.Start_Bins and FLW.End_Bins are vectors that contain what bins are summed to form the 16 bands in the FLW packet.

Parameter Name	Data Type	TC*	Description
FLP.ADC_Sample_Period	uint32	n	Number of FPGA ticks between FLP ADC sample triggers
FLP.Accumulate_Count	uint16	n	Number of FLP samples to accumulate
FLP.Digital_Gain	uint8	y	2^N gain of the accumulated value
FLP.DAC_Word	uint16	y	Output word for the Fixed Langmuir Probe

*Tele-command Configurable

Table 4.3: FLP Configuration Values

Parameter Name	Data Type	TC*	Description
FLW.Start_Bins	uint16 [16]	n	Vector of the starting FFT bins for each of the 16 spectrometer bins
FLW.End_Bins	uint16 [16]	n	Vector of the ending FFT bins for each of the 16 spectrometer bins

*Tele-command Configurable

Table 4.4: FLW Configuration Values

4.5 Sweeping Langmuir Probe (SLP)

The SLP sweeps over a range of voltages and measures currents to form a Langmuir IV curve. The SLP uses a DAC to control the voltage of the probe and an ADC to measure the current. The SLP also has a gain select line that selects between two gains for the analog amplifier circuitry. The SLP has two types of sweeps, a full sweep and a fast sweep. The sweep is set by a control block shown in figure 4.5. The control block triggers a fast sweep every SLP.Trigger_Period system ticks. Every SLP.Full_Count sweeps the controller triggers a full sweep instead. This is controlled by a state machine shown in figure 4.6.

The state machine has an external counter that sets Sweep_trigger high for one system tick every SLP.Trigger_Period system ticks. When The state machine receives a trigger, the state machine waits for the DAC to sync. When the DAC has synced, the state machine checks if the internal sweep counter has reached SLP.Full_Count. If the sweep count has reached SLP.Full_Count, the controller triggers a full sweep and resets the internal sweep counter. If the sweep count has not reached SLP.Full_Count, the controller triggers a fast sweep and increments the internal sweep counter.

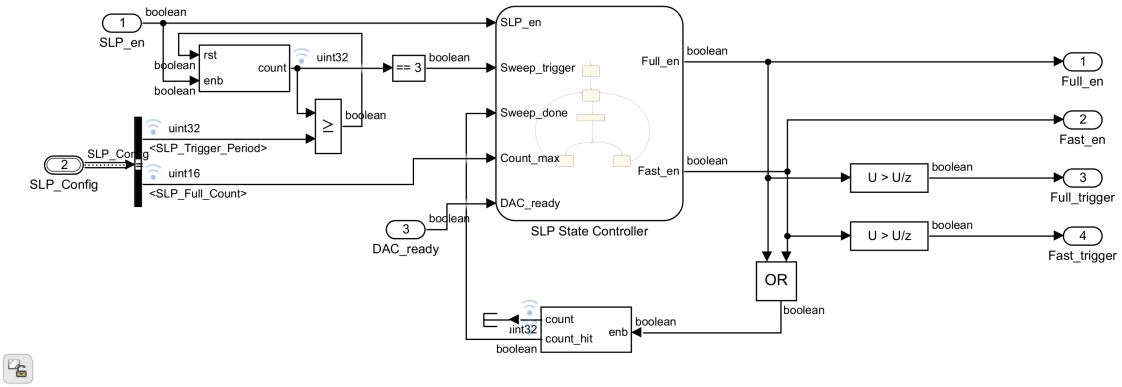

Fig. 4.5: Sweep Control Block

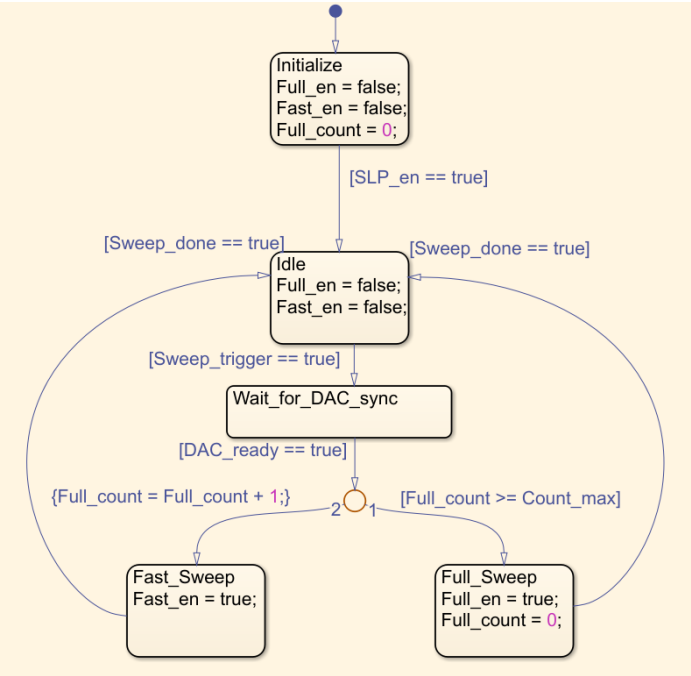

Fig. 4.6: Sweep Control State Machine

In both sweeping modes, at each sweep step, the SLP outputs a DAC word and the SLP uses the skip sum filter shown in figure 4.7 to take a current sample. The skip sum filter has an internal counter that counts to 10 samples. The first two samples are skipped, and the remaining 8 samples are accumulated using an adder and a latch. The accumulated value is then bit shifted and sliced into a 20 bit container. When the counter reaches 10, the accumulator and counter are reset. At each sweep step, an FPP value is also accumulated by an average down-sample block as seen in figure 4.2. The accumulated SLP current value and the accumulated FPP value are put into a granule. All granules from a sweep are assembled into a packet.

The SLP sweep is controlled by the parameters in table 4.5. SLP.ADC_Sample_Period controls how many system ticks between SLP samples during a sweep. The SLP.Skip, SLP.Sum, and SLP.Digital_Gain control the skip sum filter that processes the SLP data. SLP.Trigger_Period and SLP.Full_Count control the state machine. SLP.FPP_Accumulate_Count and SLP.FPP_Digital_Gain control the FPP average down-sample block.

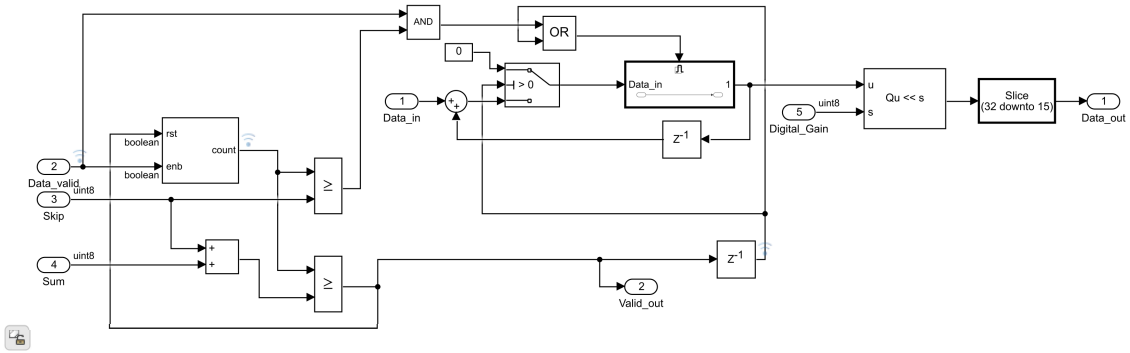

Fig. 4.7: Skip Sum Filter

Parameter Name	Data Type	TC*	Description
SLP.ADC_Sample_Period	uint32	n	Number of FPGA Ticks between SLP ADC sample triggers
SLP.Skip	uint8	n	Number of oversampled SLP samples to skip
SLP.Sum	uint8	n	Number of oversampled SLP samples to accumulate
SLP.Digital_Gain	uint8	y	2^N gain of the accumulated SLP value
SLP.Trigger_Period	uint32	y	Number of FGPA ticks between fast sweep start signals
SLP.Step_Duration	uint16	n	Number of FPGA ticks between sweep steps
SLP.Full_Count	uint16	y	After every Full_Count fast sweeps, a full sweep is performed
SLP.FPP_Accumulate_Count	uint16	n	Number of FPP samples to accumulate for the SLP packets
SLP.FPP_Digital_Gain	uint8	y	2^N gain of the accumulated FPP value

*Tele-command Configurable

Table 4.5: SLP Configuration Values

4.5.1 Fast Sweep Packet

The fast sweep is controlled by the block shown in figure 4.8. The fast sweep controller is controlled by parameters in table 4.6. When the controller receives a sweep trigger, the sweeping control state machine waits until a processed FPP sample is completed. This sample is stored and used for the sweep. When the FPP sample is taken the state machine switches to sweeping mode. In sweeping mode, a counter is enabled that rolls over

every SLP.Step_Duration system ticks When this counter rolls over, another counter is incremented, incrementing the sweep table index. The first four sweep steps use a fixed offset value of 2^{15} so that the first four sweep steps of the table are always the DAC words that correspond to $-2V$, $-1.99V$, $2.99V$, and $3V$. After the first four samples, the fast sweep then sweeps through a table of offsets from the initial value measured by the FPP. The fast sweep DAC word is calculated by equation 4.1. At each step in the table, there is a gain selection value, which is a boolean of either 0 or 1. This value selects the gain the Langmuir probe amplifier uses. Once the sweep is completed, the state goes back into idle mode and waits for another trigger. The DAC word offset and gain section value at each step is determined from the tables SLP.Fast_Offset_Table, SLP.Fast_Offset, and SLP.Fast_Gain_Table.

$$\text{DAC\_word} = \text{SLP.Fast\_Offset\_Table}[i] + \text{FPP\_value} + \text{SLP.Fast\_Offset} \tag{4.1}$$

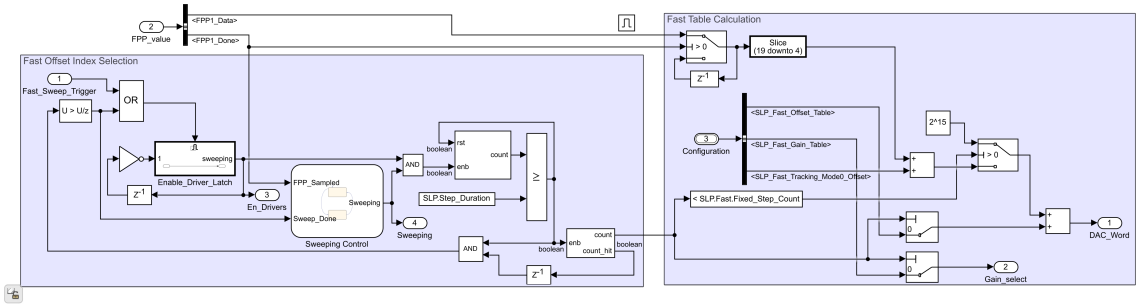

Fig. 4.8: Fast Table Sweep Controller

Parameter Name	Data Type	TC*	Description
SLP.Fast_Offset_Table	uint16 [32]	y	Table of offsets from the current track value
SLP.Fast_Gain_Table	boolean [32]	y	Table of gain selects during a fast sweep

*Tele-command Configurable

Table 4.6: Fast Packet Configuration Values

4.5.2 Full Sweep Packet

The full sweep is controlled by the block shown in figure 4.9. When the full sweep controller receives a trigger, the sweep index selection changes to sweep mode. In sweeping mode a counter is enabled that rolls over every SLP.Step_Duration system ticks When this counter rolls over, another counter is incremented, incrementing the sweep table index. The DAC word and gain section value at each step is determined from the tables SLP.Full_DAC_Word_Table and SLP.Full_Gain_Table as seen in table 4.7.

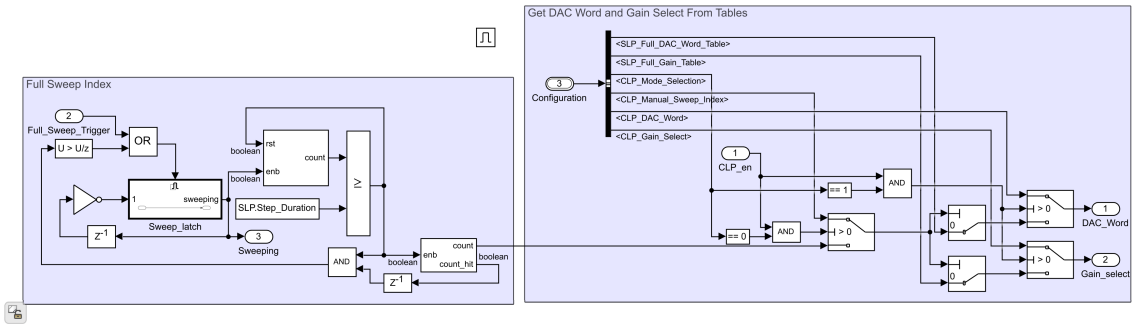

Fig. 4.9: Full Table Sweep Controller

Parameter Name	Data Type	TC*	Description
SLP.Full_DAC_Word_Table	uint16 [1051]	y	Table of words to send to the DAC in a Full sweep
SLP.Full_Gain_Table	boolean [1051]	y	Table of gain selects during a Full sweep

*Tele-command Configurable

Table 4.7: Full Packet Configuration Values

4.5.3 Calibration Mode

In calibration mode, the SLP only produces full packets. The DAC word and gain select lines are directly controlled by configurable parameters seen in table 4.8. CLP.Mode_Selection controls what calibration mode the instrument operates in. In mode 0, the DAC word and gain selection value are the full table entries with index CLP.Manual_Sweep_Index. In mode 1, the DAC word and gain selection value are set to the parameters CLP.DAC_Word and CLP.Gain_Select. The DAC word and Gain select line are controlled by using the switches seen in figure 4.9.

Parameter Name	Data Type	TC*	Description
CLP.Mode_Selection	boolean	y	0: select sweep index from full sweep table. 1: select DAC word and gain
CLP.Manual_Sweep_Index	uint16	y	Sweep index of Full sweep table for the manual sweep
CLP.DAC_Word	uint16	y	DAC word to write in calibration mode
CLP.Gain_Select	boolean	y	Select gain in calibration mode

*Tele-command Configurable

Table 4.8: SLP Callibration Configuration Values

4.6 Impedance Probe (IP)

The impedance probe has two main operational modes, science mode and calibration mode. In science mode, the IP has four different sub-modes that each produce different packet types: short sweep packets (SIP), long sweep packets (LIP), Tracking Packets (TIP), and Tracking Q-Factor packets (QIP). The packet produced is controlled by the state machine shown in figure 4.10. The control block triggers a frame every IP.Track_Trigger_Period system ticks. A frame, as shown in figure 3.10, consists of a track section and a sweep section as discussed in chapter 3. The variables IP.Q, IP.S, and IP.L correspond to the parameters Q, S, and L discussed in chapter 3.

The state machine starts in an initialize state. When the probe is enabled, the state machine enters an idle state. An external counter triggers a frame every IP.Track_Trigger_Period system ticks. When track is triggered, the state machine enables the tracking mode for the TIP and QIP. After TIP.Sample_Period system ticks, the state machine checks if a QIP sweep counter has reached the IP.Q value. If the counter has reached IP.Q, the QIP sweep is enabled and the QIP sweep counter is reset. Otherwise, a TIP sample is taken and the QIP sweep counter is incremented. The state machine then waits — system ticks for the

QIP sweep to complete. The state machine then checks if a SIP sweep counter has reached IP.S. If the counter has not reached IP.S, the state machine returns to the idle state and the SIP sweep counter is incremented. Otherwise, the state machine checks if a LIP sweep counter has reached IP.L. If the LIP sweep counter has not reached IP.L, an SIP sweep is triggered, the SIP sweep counter is reset, and the LIP sweep counter is incremented. Otherwise, a LIP sweep is triggered, and the SIP sweep counter and the LIP sweep counter are reset. After a SIP sweep or LIP sweep is triggered, the state machine returns to the idle state.

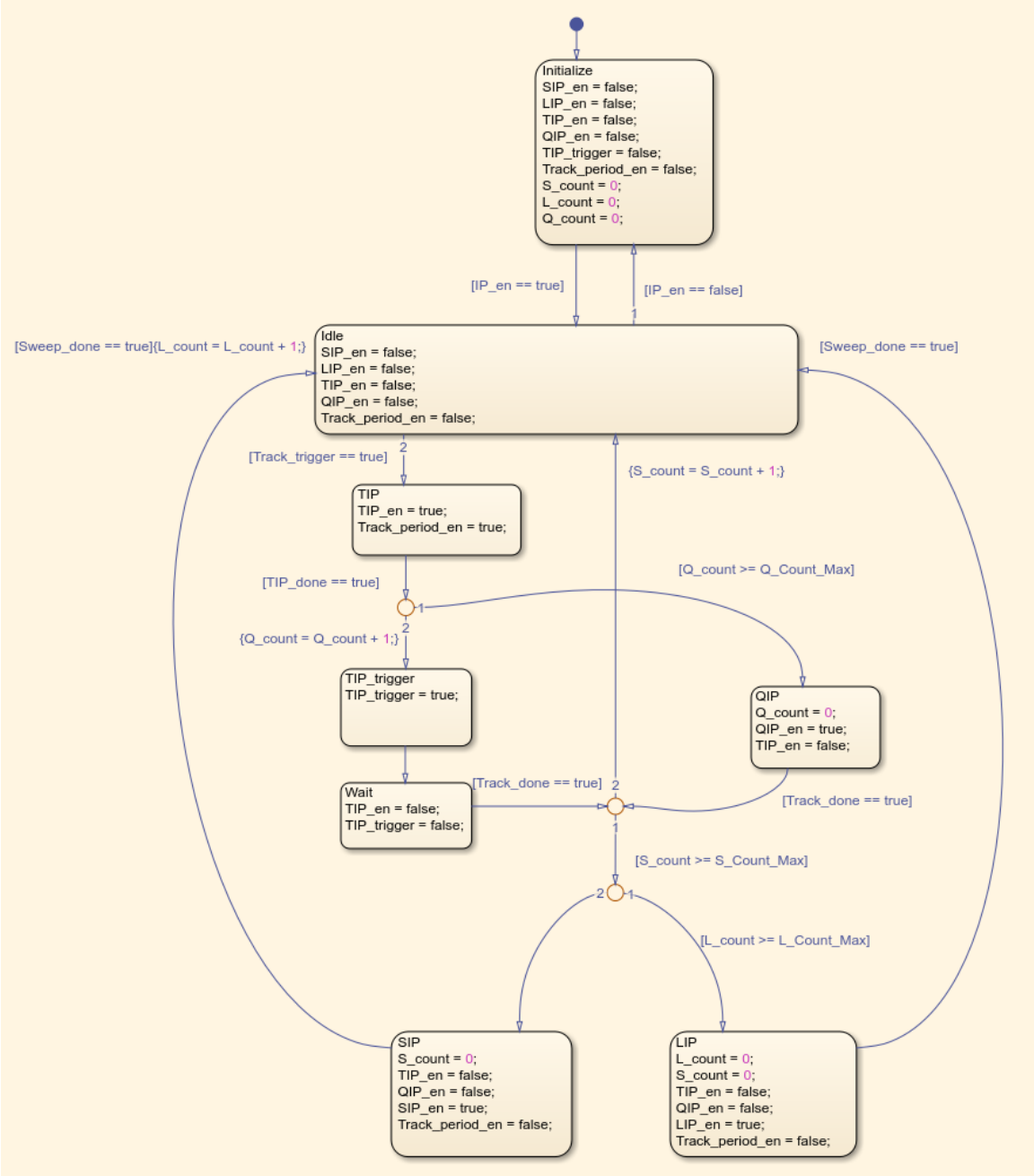

Fig. 4.10: IP Control State Machine

The impedance probe processing consists of several blocks as shown in figure 4.11. The impedance probe has a frequency word selector that creates a 20-bit frequency word. The frequency word selector operates differently depending on the mode of operation. The

frequency word controls the oscillators which generate a local cosine and sine wave at the specified frequency for I and Q calculation as well as an output cosine at the specified frequency. The amplitude of the output cosine is controlled by an amplitude DAC word. The amplitude DAC word is controlled by a calibration table that is configured in instrument calibration. The phase of the local signals is controlled by a phase calibration table. The input from the probe and the local oscillator signals are mixed to calculate the I and Q values at the given frequency. In tracking mode, the phase of the I and Q values is calculated and used in a phase tracking loop to control the frequency word.

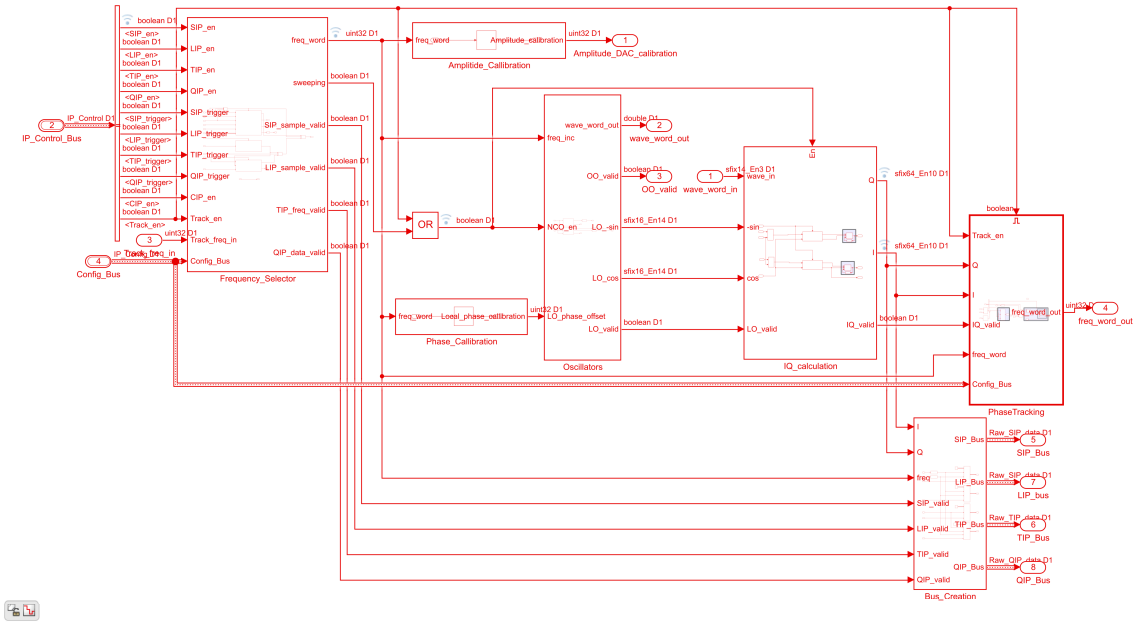

Fig. 4.11: IP Processing

The oscillators for the impedance probe are shown in figure 4.12. Depending on the mode of operation, a frequency word is selected. This word represents the frequency the numerically controlled oscillators (NCO) operate at. The frequency word is a 20-bit unsigned number that corresponds to a given frequency by equation 4.2. F_0 is the desired frequency in Hz, T_s is the sample time, $T_s = 6.25ns$, and N is the NCO accumulator length, $N = 20$. The frequency resolution of the NCO is given by equation 4.3 where Δf is the frequency

resolution in Hz.

$$word = round(F_0 T_s 2^N) \tag{4.2}$$

$$\Delta f = \frac{1}{T_s 2^N} = 152.5879 Hz \tag{4.3}$$

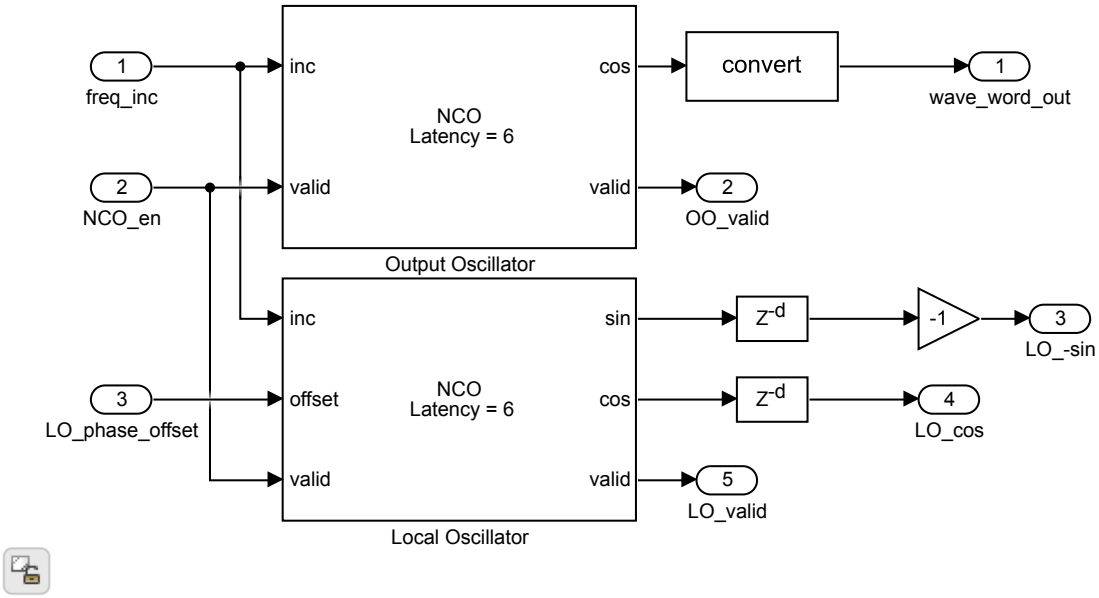

Fig. 4.12: Oscillators

The I and Q samples are calculated in the block shown in figure 4.13. The I and Q values are calculated by mixing the incoming signal from the probe with the locally generated cosine and sine wave. The mixed signal is then lowpass filtered by using the average downsample block used in the FPP probe. The output of the average downsample block is latched. The final I and Q values are 20-bit values.

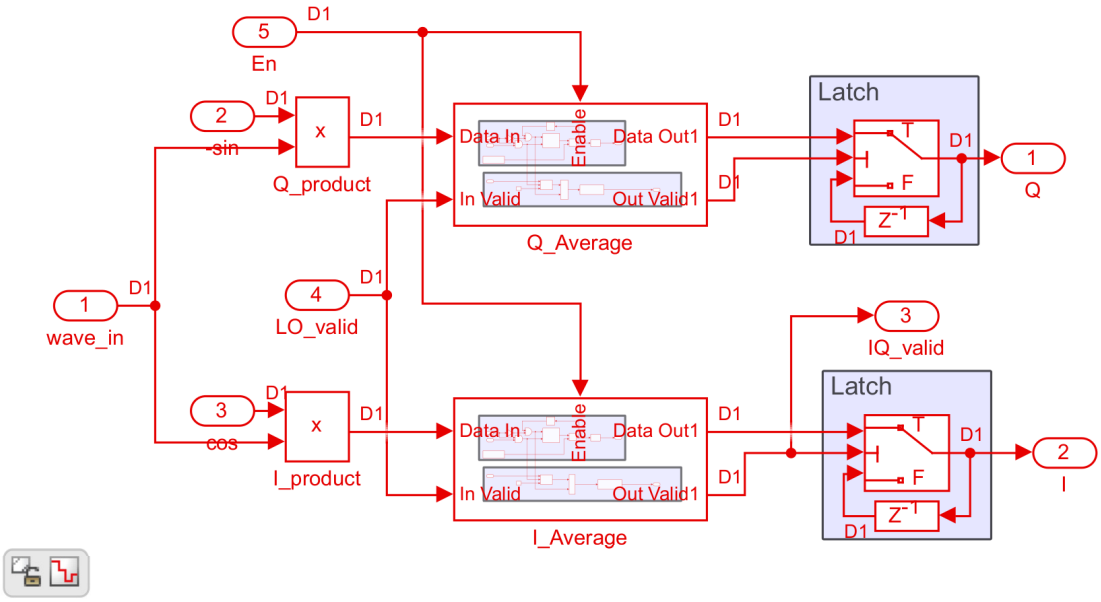

Fig. 4.13: IQ Calculation

Parameter Name	Data Type	TC*	Description
IP.Track_Trigger_Period	uint32	n	Number of FPGA ticks between track sample triggers
IP.Sweep_Period	uint32	n	Number of FPGA ticks in an long frequency sweep
IP.Track_Period	uint32	n	Number of FPGA ticks to track to a frequency and sweep through all frequency offsets
IP.Q	uint16	y	Every Q Track triggers, a QIP packet is produced instead of a TIP
IP.S	uint16	y	Every S track triggers, a Sweep is performed after the track (default SIP packet)
IP.L	uint16	y	Every L sweeps, a LIP packet is created instead of a SIP
IP.Coarse_Delay	uint16	n	Number of FPGA ticks to delay the Local Oscillator outputs
IP.DAC_Amplitude _Calibration_Table	uint16 [600]	n	Table of DAC amplitude words to send at each frequency step
IP.Phase_Calibration_Table	uint16 [600]	n	Table of phase adjustments at each frequency step
IP.Sweep_Freq_Table	uint16 [600]	n	Table of frequency steps for sweeping mode

*Tele-command Configurable

Table 4.9: IP General Configuration Values

4.6.1 SIP/LIP packets

The operation of the frequency selector for SIP and LIP packets is shown in figure 4.14. When a sweep is triggered, the corresponding index selection block will enter sweeping state. This enables a counter that rolls over every SIP.Freq_Step_Duration system ticks. When the counter rolls over, another counter is triggered that increments the sweep index. In an SIP packet sweep, the index selection block sweeps over the top 512 indices of the IP.Sweep_Freq_Table. In an LIP packet sweep, the index selection block sweeps over the full IP.Sweep_Freq_Table. At each frequency step, the I and Q values are assembled into the SIP and LIP packets.

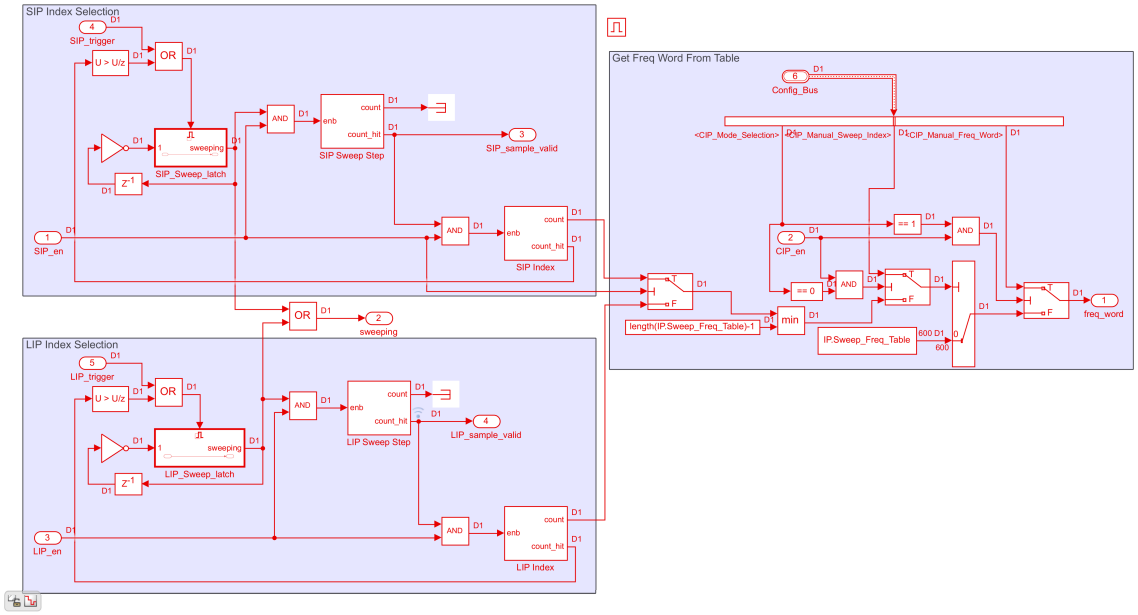

Fig. 4.14: SIP/LIP Frequency Selection

Parameter Name	Data Type	TC*	Description
SIP.Freq_Step_Duration	uint32	n	Number of FPGA ticks between each sweep step

*Tele-command Configurable

Table 4.10: SIP Configuration Values

4.6.2 TIP packet

When the impedance probe is creating a TIP packet, the frequency word is controlled by a tracking loop as seen in figure 4.15. The parameters in table 4.11 control the tracking loop of the probe. The loop calculates the phase from the I and Q samples calculated above. The phase is calculated using the CORDIC technique. The phase error is calculated from the TIP.Track_Phase_Table and the error is sent into a discrete time PI loop filter. The transfer function of the loop filter is shown in equation 4.4 where T_s is the sample time $T_s = 6.25ns$ and K_p and K_i are filter gains given by TIP.Track_Gains. The output of the loop filter is then accumulated to form a frequency word. The frequency word is limited by the values in TIP.Freq_Limits. When the impedance probe begins a track, the frequency word the tracking loop begins at is TIP.Initial_Freq. TIP.Sample_Period system ticks after the impedance probe enters tracking mode in a TIP track, the tracking frequency word is latched and put into a TIP granule.

$$H(z) = K_p + \frac{K_i T_s}{z - 1} \tag{4.4}$$

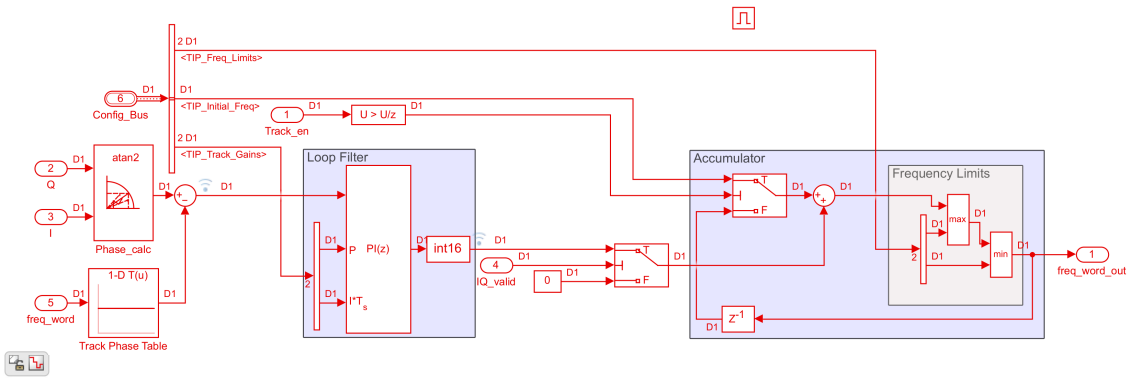

Fig. 4.15: Phase Tracking Loop

Parameter Name	Data Type	TC*	Description
TIP.Freq_Limits	uint32 [2]	y	Frequency word limits for the phase tracking
TIP.Initial_Freq	uint32	y	Initial frequency word for phase tracking
TIP.Track_Gains	fixdt(1,32,14) [2]	y	Gains to use in the PI controller
TIP.Track_Phase_Table	fixdt(1,16,14) [600]	n	Table of phases to track to at each frequency
TIP.Sample_Period	uint32	n	Number of system ticks between a TIP/QIP track begins and a sample is taken

*Tele-command Configurable

Table 4.11: TIP Configuration Values

4.6.3 QIP packet

The QIP packet operates in a similar manner to the TIP packet. The only difference is after TIP.Sample_Period system ticks the frequency word is latched and the impedance probe sweeps through frequencies around the latched frequency. The QIP sweep con-

troller works like the SIP and the LIP sweep using two counters to iterate through the QIP.Freq_Offset_Table, incrementing one index every QIP.Freq_Offset_Period system ticks. The frequency at each sweep step is QIP.Freq_Offset_Table[i] plus the latched frequency. The latched frequency and the I and Q values at each QIP.Freq_Offset_Table step are assembled into a QIP granule.

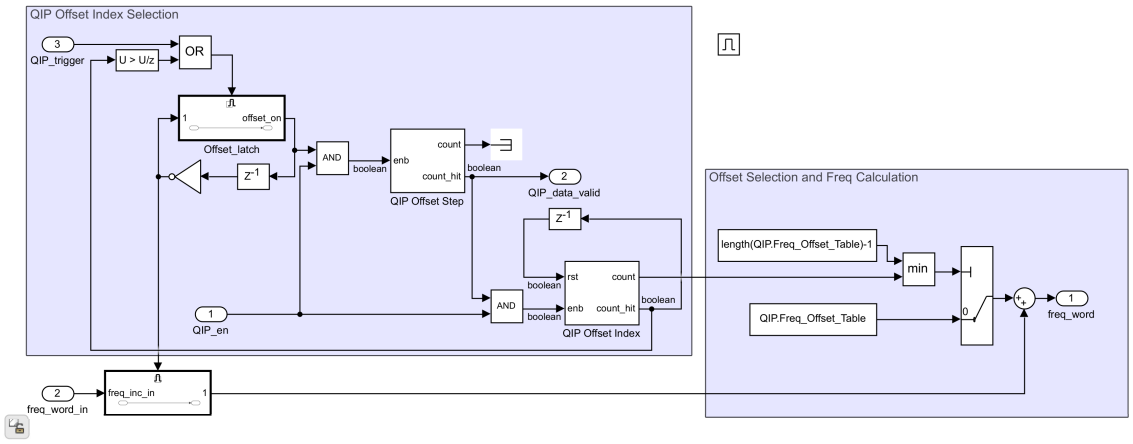

Fig. 4.16: QIP Offset Calculation

Parameter Name	Data Type	TC*	Description
QIP.Freq_Offset_Table	int32	n	Table of frequency offsets in QIP
QIP.Freq_Offset_Period	uint16	n	Number of FPGA ticks between each QIP table step

*Tele-command Configurable

Table 4.12: QIP Configuration Values

4.6.4 Calibration mode

In calibration mode, the impedance probe creates SIP packets. There are two modes in calibration mode. CIP.Mode_Selection selects the calibration mode. In mode 0, the frequency word is held at the index of the sweep table given by CIP.Manual_Sweep_Index. In

mode 1, the frequency word is set to the value given by CIP.Manual_Freq_Index. The calibration mode controls the frequency word using the switches in figure 4.14. The calibration mode is used to create the tables IP.DAC_Amplitude _Calibration_Table, IP.Phase_Calibration_Table, and TIP.Track_Phase_Table.

Parameter Name	Data Type	TC*	Description
CIP.Mode_Selection	boolean	y	0 is sweep index from sweep frequency table, 1 is manually entered frequency word
CIP.Manual_Sweep_Index	uint16	y	Sweep index for the manual sweep
CIP.Manual_Freq_Index	uint32	y	Manually set Frequency Word

*Tele-command Configurable

Table 4.13: IP Calibration Configuration Values

CHAPTER 5

Results

This chapter outlines the results from simulating the behavior of each of the probes. The methods used for the simulations are outlined. The number of simulations for each probe is limited due to the length of time required to run a simulation.

5.1 Floating Potential Probe

The testing results for the floating potential probe are shown below.

5.1.1 FPP Packet

To test the FPP Packet, four test signals were injected into the model, sampled at a 200 kHz rate one for each channel. The signals are shown in equation 5.1 where $f = 10$ and $\phi_i = 0, \frac{\pi}{8}, \frac{\pi}{4}, \frac{3\pi}{8}$ so that each signal is $\frac{\pi}{8}$ radians out of phase with each other. The input signals are shown in figure 5.1. The FPP lowpass filters the data and applies a gain to produce an output signal at 100 Hz. The output of the FPP is compared to the input data in figure 5.2. The output data has the same frequency as the input data. The filter has a phase delay as seen in figure 5.2, however the relative phases of the four signals is preserved.

$$s(t) = cos(2\pi ft + \phi_i) \tag{5.1}$$

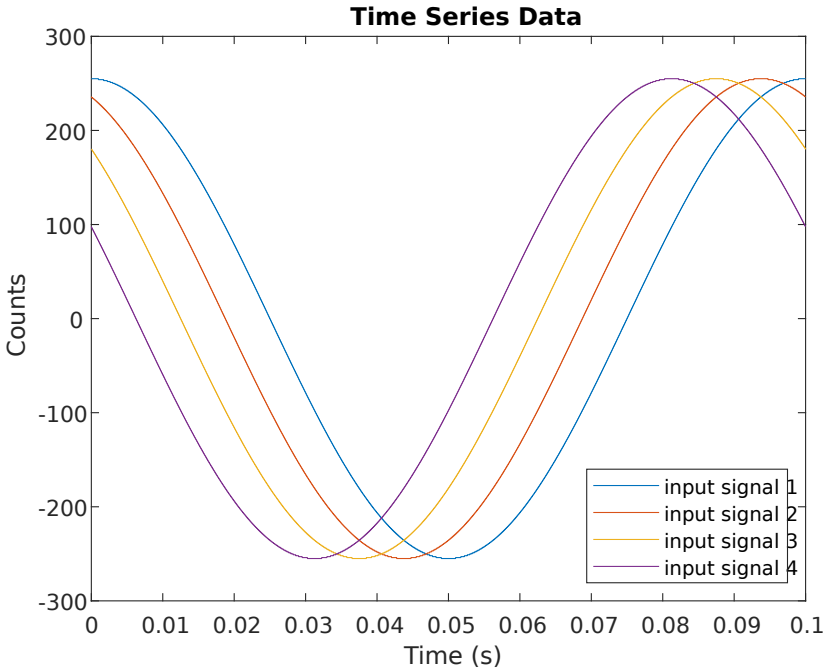

Fig. 5.1: FPP Input Data

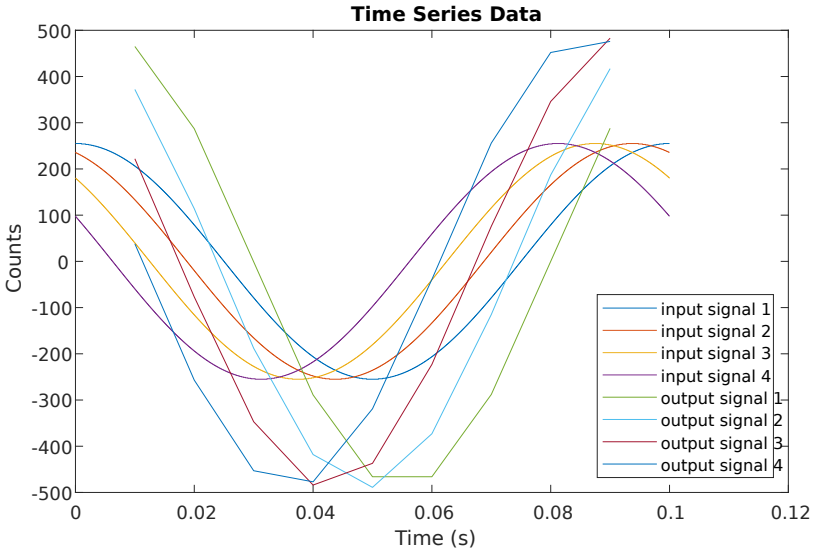

Fig. 5.2: FPP Input and Output Data

5.1.2 Measurement Requirements

The FPP meets the sample time requirements form table 3.1. The measured sample rate is shown in table 5.1.

Parameter	Required Value	Actual Value
Voltage Measurement	100 Hz	100 Hz

Table 5.1: FPP Measurement Requirements

5.2 Electric Field Wave Spectrometer

The testing results for the electric field wave spectrometer are shown below.

5.2.1 EFW Packet

The EFW packet is tested by injecting three different white gaussian noise signals into the design. The injected signals have different power levels, $0dB$, $3dB$, and $6dB$ as seen in figure 5.3. The EFW calculates the PSD of the input signal. The PSD of an ideal white noise is constant for all frequencies and the amplitude of the PSD is proportional to the power of the noise signal. The simulated noise signal is not an ideal noise signal, so the PSD is not perfectly constant. The frequency bands the EFW produces are shown in table 5.2. The frequency band outputs from the EFW are shown in figure 5.4.

Frequency Band	FFT Bins	Frequency Range
1	1 - 50	0 - 4.7852 kHz
2	51 - 100	4.8828 - 9.668 kHz
3	101 - 150	9.7656 kHz - 14.551 kHz
4	151 - 200	14.648 kHz - 19.434 kHz
5	201 - 250	19.531 kHz - 24.316 kHz
6	251 - 300	24.414 kHz - 29.199 kHz
7	301 - 350	29.297 kHz - 34.082 kHz
8	351 - 400	34.18 kHz - 38.965 kHz
9	401 - 450	39.063 kHz - 43.848 kHz
10	451 - 500	43.945 kHz - 48.73 kHz
11	501 - 550	48.828 kHz - 53.613 kHz
12	551 - 600	53.711 kHz - 58.496 kHz
13	601 - 650	58.594 kHz - 63.379 kHz
14	651 - 700	63.477 kHz - 69.262 kHz
15	701 - 750	68.359 kHz - 73.242 kHz
16	751 - 800	73.242 kHz - 78.027 kHz

Table 5.2: EFW Frequency Bands

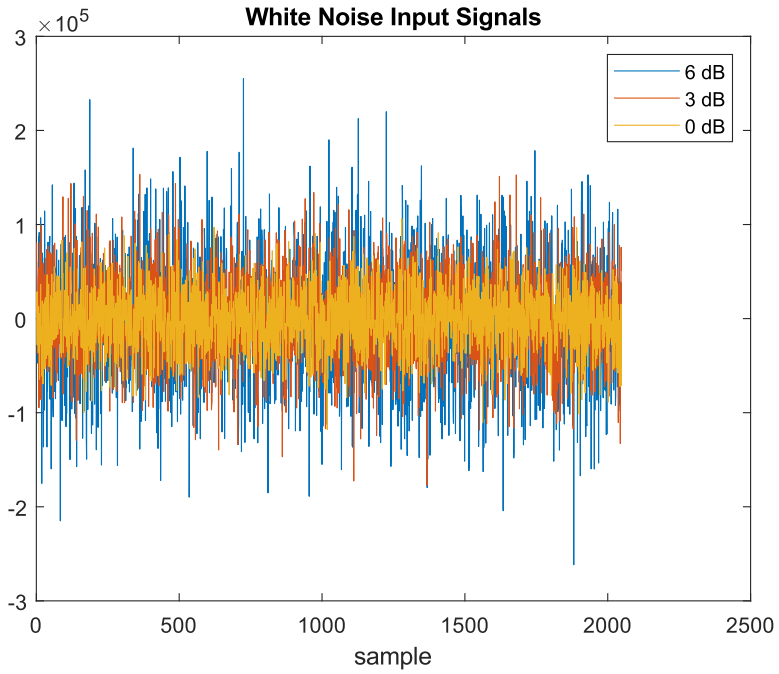

Fig. 5.3: EFW Input Data

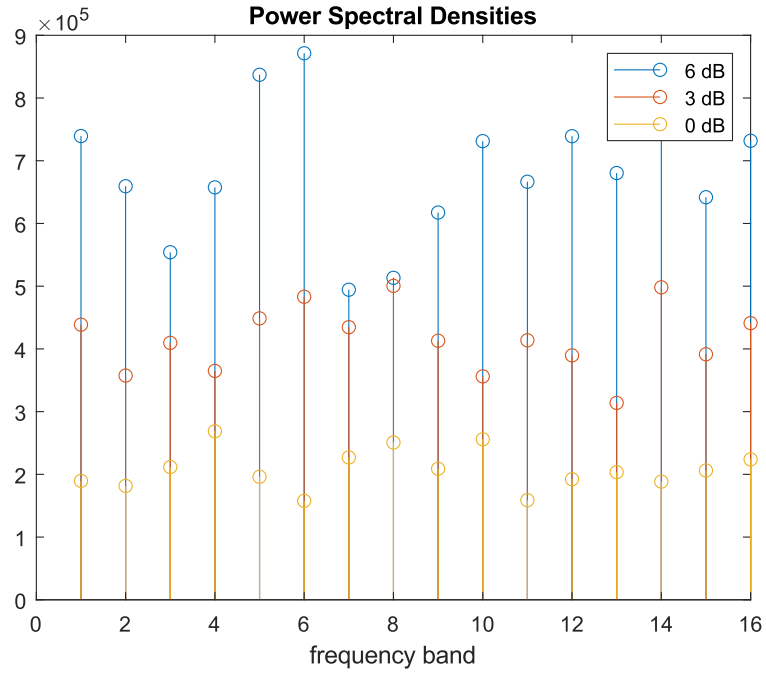

Fig. 5.4: EFW PSD Output

5.2.2 Measurement Requirements

The EFW spectrograms meet the sample time requirements form table 3.1. The measured sample rate is shown in table 5.3.

Parameter	Required Value	Actual Value
Current Spectrogram	10 Hz	12 Hz

Table 5.3: EFW Measurement Requirements

5.3 Fixed Bias Langmuir Probe

The FLP was successfully simulated in MATLAB/Simulink. The SLP performed as expected and met the sample rate requirements.

The fixed bias Langmuir probe was simulated in two different ways, one fore each packet type the probe produces. To simulate the FLP Packet, the same waveform used to simulate the FPP was injected into the FLP, since the FLP packet uses the same processing as the FPP. To simulate the FLW Packet, the same set of waveforms used to simulate the EFW were injected into the FLP, since the FLW packet uses the same processing as the EFW.

5.3.1 FLP Packet

To test the FLP packet, the FLP was injected with the waveform shown in figure 5.5. This is a sinusoid with a frequency of 10 Hz. The FLP performs a lowpass filter on the signal and applies a gain. The gain applied by the FLP is different than the FPP gain due to a different down-sampling rate. The output of the FLP is comapred to the input in figure 5.6. The output frequency is the same as the input frequency and the only differences are a gain from the FLP, and a phase shift from the low pass filter.

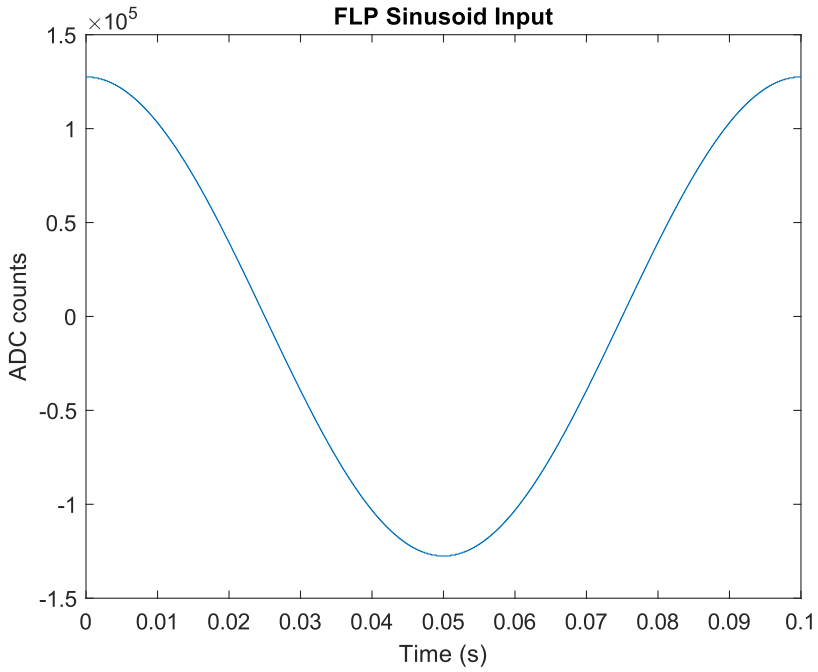

Fig. 5.5: FLP Packet Input Data

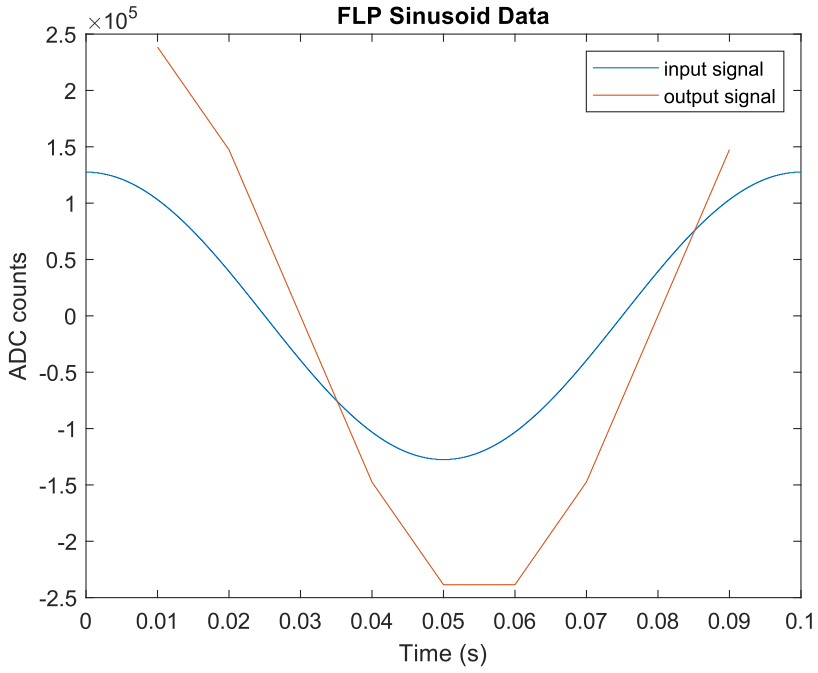

Fig. 5.6: FLP Packet Results

5.3.2 FLW Packet

The FLW packet is tested by injecting three different white gaussian noise signals. The injected signals have different power levels, $0dB, 3dB,$ and $6dB$ as seen in figure 5.7. These signals are the same as the signals used to test the EFW, since the FLW packet uses the same processing as the EFW probe. The resulting frequency bands are shown in table 5.4. The resulting PSD from the FLW is shown in figure 5.8.

Frequency Band	FFT Bins	Frequency Range
1	1 - 50	0 - 2.3926 kHz
2	51 - 100	2.4414 - 4.834 kHz
3	101 - 150	4.8828 kHz - 7.2754 kHz
4	151 - 200	7.3242 kHz - 9.7168 kHz
5	201 - 250	9.7656 kHz - 12.158 kHz
6	251 - 300	12.207 kHz - 14.6 kHz
7	301 - 350	14.648 kHz - 17.041 kHz
8	351 - 400	17.09 kHz - 19.482 kHz
9	401 - 450	19.531 kHz - 21.924 kHz
10	451 - 500	21.973 kHz - 24.365 kHz
11	501 - 550	24.414 kHz - 26.807 kHz
12	551 - 600	26.855 kHz - 29.248 kHz
13	601 - 650	29.297 kHz - 31.689 kHz
14	651 - 700	31.738 kHz - 34.131 kHz
15	701 - 750	34.18 kHz - 36.572 kHz
16	751 - 800	36.621 kHz - 39.014 kHz

Table 5.4: FLW Frequency Bands

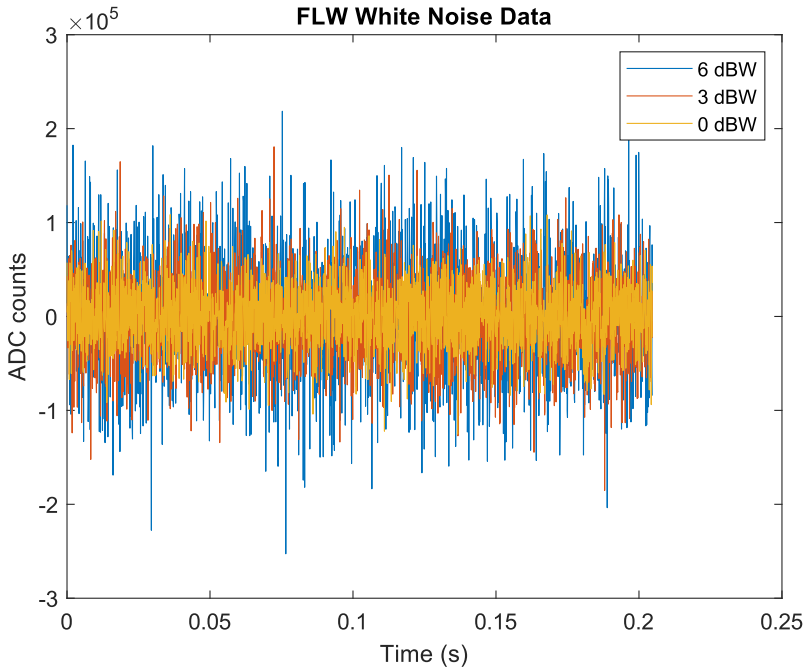

Fig. 5.7: FLW Input Data

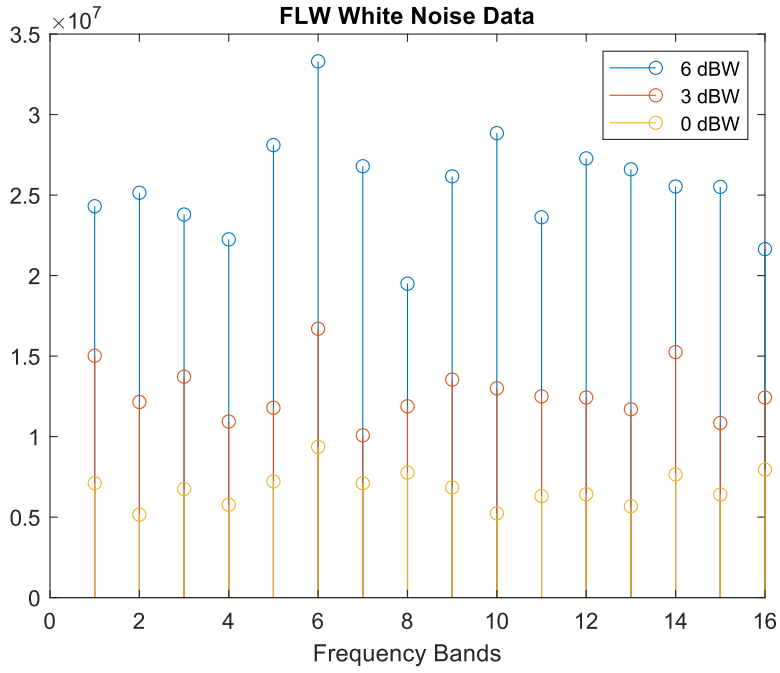

Fig. 5.8: FLW PSD Output

5.3.3 Measurement Requirements

The FLP meets the sample time requirements form table 3.1. The actual sample rates are shown in table 5.5.

Parameter	Required Value	Actual Value
Current Measurement	100 Hz	100 Hz
Current Spectrogram	10 Hz	12 Hz

Table 5.5: FLP Measurement Requirements

5.4 Sweeping Langmuir Probe

The SLP was successfully simulated in MATLAB/Simulink. The SLP performed as expected and met the sample rate requirements.

To test the SLP, a probe simulation block was created. The probe simulation simulates the DAC by converting the DAC word to a voltage. The simulation then uses a look up table to simulate the Langmuir IV curve shown in figure 5.9. The simulation then converts the current to the equivalent ADC count. The ADC count is then read by the digital design. The IV curve used for the simulation is shown in figure 5.10.

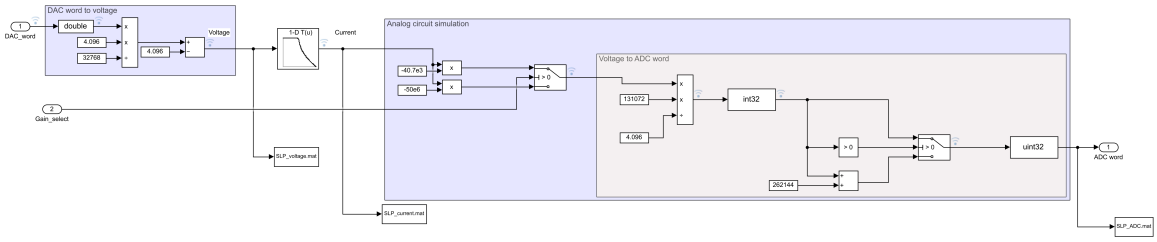

Fig. 5.9: SLP Probe Simulation

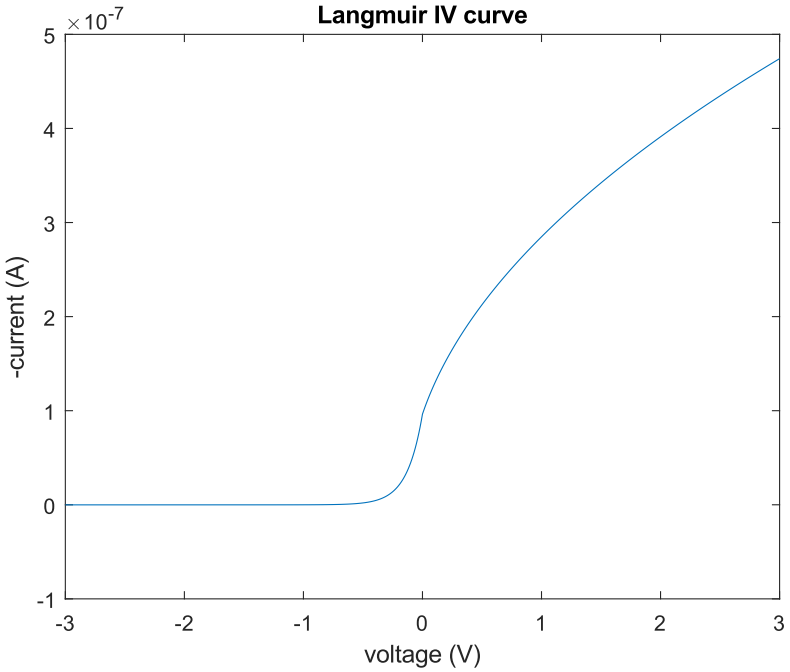

Fig. 5.10: SLP IV Curve Simulation

The SLP simulation holds the FPP value at $-0.97?V$, or 100032 as a DAC word.

5.4.1 Full Packet

The results for the full packet are shown below.

Simulation Tests

The full voltage sweep and the current measured at each step is shown in figure 5.11. The Langmuir IV curve can be seen in figure 5.11. The samples read from the full packet are compared the Langmuir IV curve in figure 5.12.

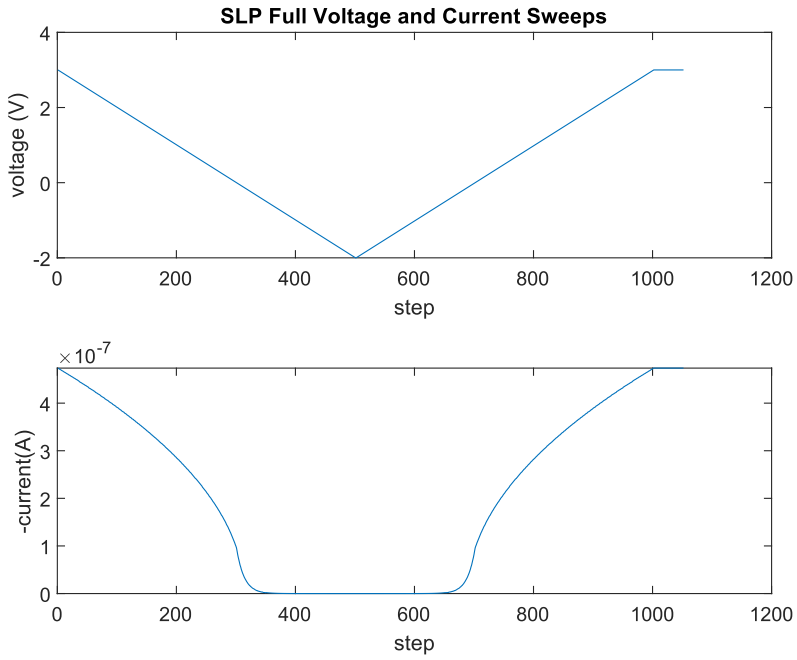

Fig. 5.11: SLP Full Packet Sweep Test Results

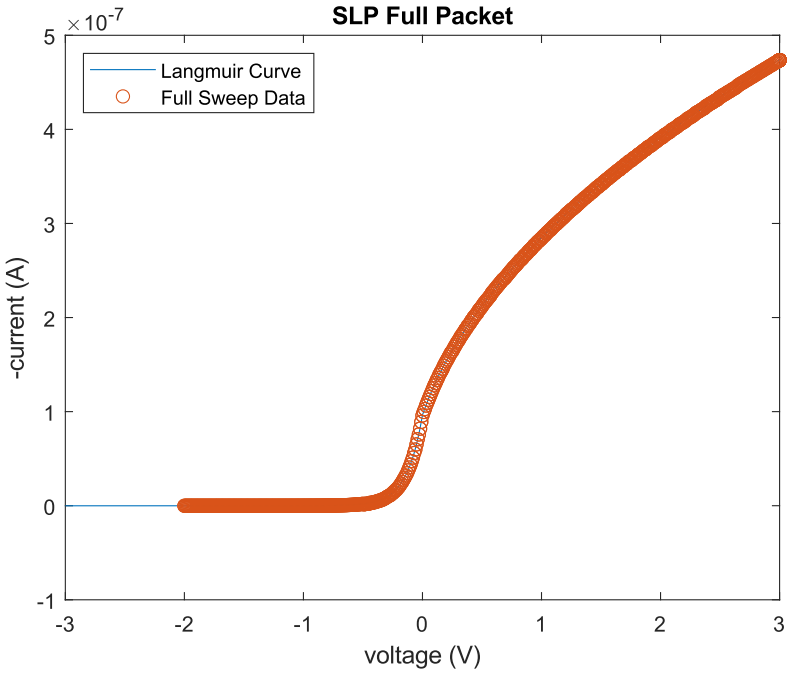

Fig. 5.12: SLP Full Packet Test Results

FPP values

The FPP value read by the SLP in the full packet is 100032 ADC counts, or $-0.97V$.

5.4.2 Fast Packet

The results for the fast packet are shown below.

Simulation Tests

The variable portion of the fast curve is centered around the voltage read by the FPP. The voltage sweep of the fast packet is compared to the current measured in figure 5.13. The shape of the IV curve can be seen in figure 5.13. The samples read from the fast packet are compared the Langmuir IV curve in figure 5.14.

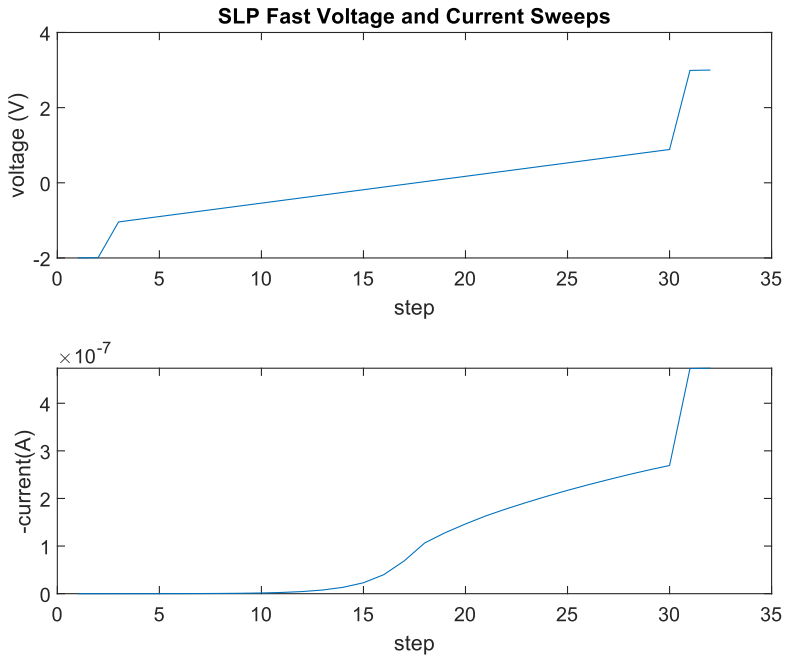

Fig. 5.13: SLP Fast Packet Sweep Test Results

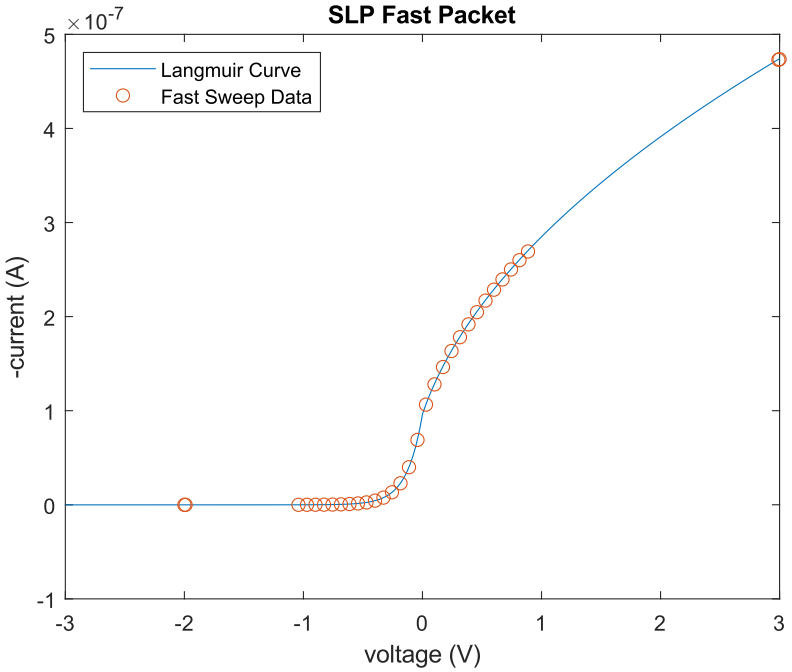

Fig. 5.14: SLP Fast Packet Test Results

FPP values

The FPP value read by the SLP in the fast packet is 100032 ADC counts, or $-0.97V$.

5.4.3 Measurement Requirements

The SLP sweeps meet the sample time requirements form table 3.1. The actual sample rates are shown in table 5.6.

Parameter	Required Value	Actual Value
Voltage Sweep	1 kHz	20 kHz
Current Measurement	10 kHz	10 kHz
Langmuir Sweep	1 Hz	1.6 kHz

Table 5.6: SLP Measurement Requirements

5.5 Impedance Probe

To test the IP, a probe simulation block was created, seen in figure 5.15. The DAC word is converted from a discrete time value to a continuous time value. The continuous time value is sent to a transfer function $H(s)$ shown in figure **??**. The output of the transfer function is converted to discrete time by a sample and hold block. The discrete time output is sent to the digital design. The transfer function used has a cyclotron frequency, Ω_c at $1MHz$, and an upper hybrid frequency, ω_{uh}, at $10MHz$.

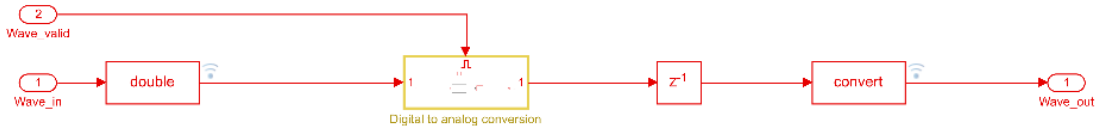

Fig. 5.15: IP Probe Simulation

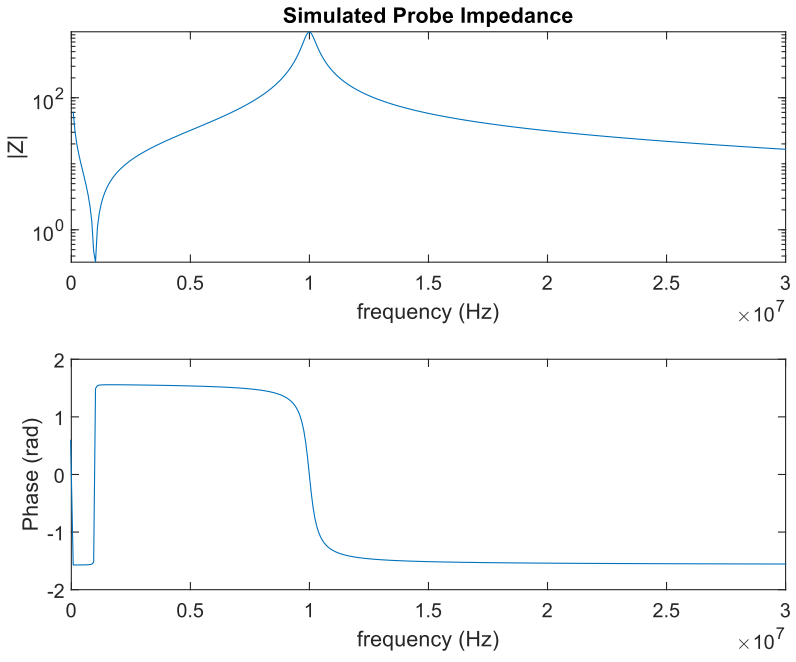

Fig. 5.16: IP Transfer Function Simulation

5.5.1 SIP Packet

The SIP packet sweeps over the frequencies from $2MHz$ to $30MHz$. The I and Q values of the impedance at each frequency step are converted into magnitude and phase as seen in figure 5.17. The impedance curve in figure 5.17 has the same shape and resonant frequencies as the simulation transfer function.

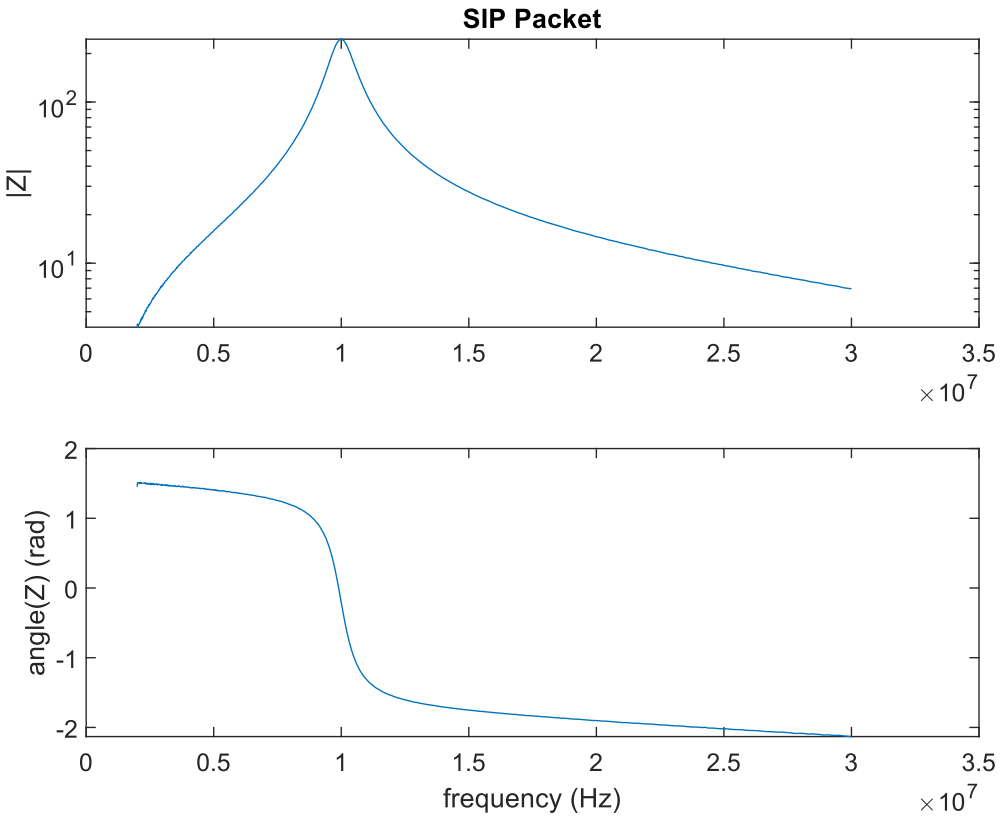

Fig. 5.17: SIP Packet Results

5.5.2 LIP Packet

The SIP packet sweeps over the frequencies from $0.5MHz$ to $30MHz$. The I and Q values of the impedance at each frequency step are converted into magnitude and phase as seen in figure 5.17. The impedance curve in figure 5.17 has the same shape and resonant frequencies as the simulation transfer function.

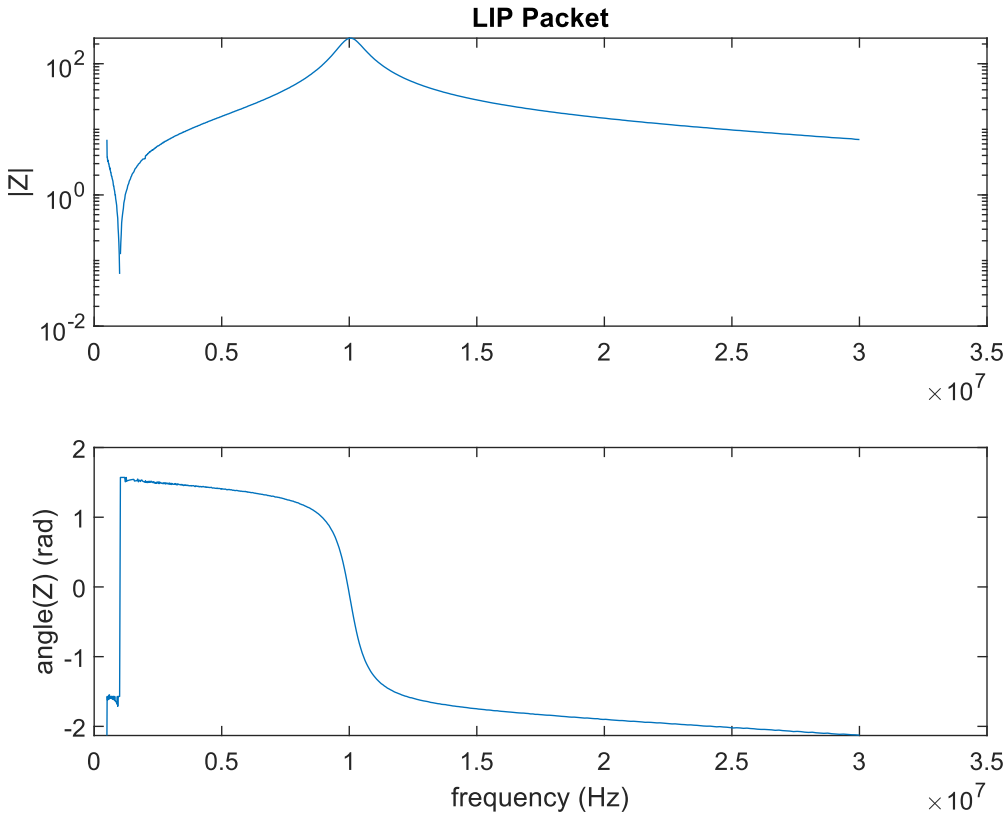

Fig. 5.18: LIP Packet Results

5.5.3 TIP Packet

The TIP produces readings of the upper resonant frequency of the impedance. The resonant frequency of the test transfer function is $10MHz$. The frequencies from the TIP are shown in table 5.7. The TIP was able to read the resonant frequency with an accuacy of approximately 0.3%.

Frequency Word	Frequency (MHz)	Percent Error (%)
65340	9.9701	0.2991
65332	9.9689	0.3113
65335	9.9693	0.3067
65335	9.9693	0.3067
65333	9.9690	0.3098
65337	9.9696	0.3036
65333	9.9690	0.3098
65334	9.9692	0.3082
65334	9.9692	0.3082
65333	9.9690	0.3098

Table 5.7: TIP Frequencies

5.5.4 QIP Packet

Th QIP packet produces readings of the upper resonant frequency of the impedance as well as the I and Q samples of the resonant peak, as well as I and Q samples of the frequencies at offsets $[-1MHz, -0.5MHz, 0.5MHz, 1MHz]$. These values can be used to calculate the Q-factor or the resonant peak. The results from the simulation are shown in figure 5.19.

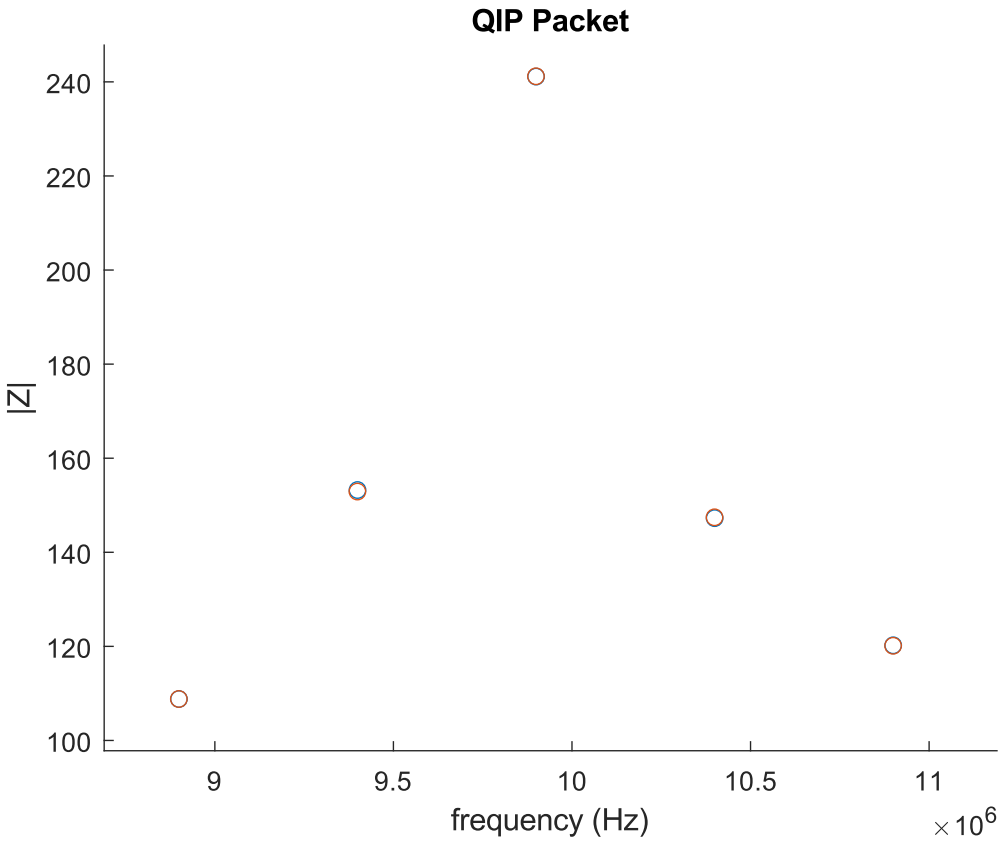

Fig. 5.19: QIP Packet Results

5.6 HDL Generation

All probe models were successfully sent into HDL coder and HDL code was generated. The HDL code for the FPP, EFW, FLP, and SLP were successfully synthesized and deployed onto the PolarFire FPGA. TThe design for the IP was successfully synthesized in the Libero environment. The design for the IP is the most complicated probe design in SWP2, and the resource utilization is shown in table 5.8.

4LUT	Count	20935
	FPGA Percentage	19.27716%
DFF	Count	7591
	FPGA Percentage	6.98987%

Table 5.8: IP resource utilization

CHAPTER 6

Conclusion

The DSP for SWP2 was successfully developed in the MATLAB/Simulink environment. The digital design and the DSP for each of the probes were successfully tested in the MATLAB/Simulink environment. All MATLAB scripts and Simulink models are in a MATLAB project linked with git, allowing USU to easily maintain the project. The probes operated as expected and met the sample rate requirements. This thesis shows that the MATLAB/Simulink environment and the HDL Coder toolchain are valid tools to create a DSP design for deployment to an FPGA. The impedance probe successfully creates a QIP packet which can be used to calculate the Q-factor of the upper hybrid resonant peak in an impedance curve.

6.1 Future work

Future work for this project includes calculating the Q-factor from the QIP packet data, tuning the probes with real data, and moving to a newer version of Simulink.

The Q-factor was not calculated from the QIP packet data in this thesis. Some preliminary work was done trying to estimate the Q-factor by interpolating between the points in the QIP packet and calculating the Q-factor of that curve. This approach did not work. To accurately calculate the Q-factor from the QIP packet, one approach could be finding the parallel LRC circuit transfer function that best matches the data points from the QIP packet. The Q-factor of the QIP packet would be the Q-factor of the parallel LRC circuit transfer function.

The IP has configurable parameters that need to be tuned according to real plasma data. This includes the tracking loop PID controller PI values which need to be tuned to accurately track the upper resonant condition. The QIP frequency offset table needs to be chosen to best calculate the Q-factor. If the offsets are too close or too far, the Q-factor

cannot be calculated.

The DSP for SWP2 was designed in version 2022a of MATLAB/Simulink. Updating to 2023a would add new functionality such as improved signal conversion blocks and custom library blocks.

REFERENCES

[1] C. Young, "Design of a sweeping impedance probe for the sport mission," Master's thesis, Utah State University, Logan, UT, 2020. [Online]. Available: https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=9022&context=etd

[2] N. Tipton, "Design of miniaturized sweeping langmuir probe and electric field probe for the sport mission," Master's thesis, Utah State University, Logan, UT, 2021. [Online]. Available: https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=9214&context=etd

[3] V. L. Pisacane, *The Space Environment and Its Effects on Space Systems, Second Edition*, 2nd ed.

[4] T. Bekkeng and E. Helgeb, "Multi-needle langmuir probe system for electron density measurements and active spacecraft potential control on cubesats," *IEEE transactions on aerospace and electronic systems*, vol. 55, no. 2, 2019.

[5] H. M. Mott-Smith and I. Langmuir, "The theory of collectors in gaseous discharges," *Physical Reviews*, vol. 2, p. 727–763, 1926.

[6] M. Kanal, "Theory of current collection of moving spherical probes," *Space Physics Research Laboratory, University of Michigan*, 1962.

[7] W. Hoegy and L. Wharton, "Current to moving spherical and cylindrical electrostatic probes." 1971.

[8] ——, "Current to a moving cylindrical electrostatic probe," *Journal of Applied Physics*, vol. 44, p. 5365–5371, 1973.

[9] J. G. Laframboise and J. Rubinstein, "Theory of a cylindrical probe in a collisionless magnetoplasma," *Physics of Fluids*, vol. 19, p. 1900–1908, 1976.

[10] J. Rubinstein and J. G. Laframboise, "Theory of a spherical probe in a collisionless magnetoplasma," *Physics of Fluids*, vol. 25, p. 1174–1182, 1982.

[11] J. E. allen, "Probe theory the orbital motion approach," *Physica Scripta*, vol. 45, 1992.

[12] F. F. Chen, "Langmuir probes in rf plasma: surprising validity of oml theory," *Plasma Sources Science and Technology*, vol. 18, 2009.

[13] F. F. C. et al, *Electric probes, in In Plasma Diagnostic Techniques.* Academic Press, 1965.

[14] E. C. Whipple, "Potentials of surfaces in space," *Reports on Progress in Physics*, vol. 44, p. 1197–1250, 1981.

[15] A. J. Auman, "Contamination free sounding rocket langmuir probe," Master's thesis, Utah State University, 2008. [Online]. Available: https://digitalcommons.usu.edu/etd/165

[16] S. Ranvier and M. Anciaux, "Use of a langmuir probe instrument on board a pico-satellite," *IEEE Transaction on Plasma Science*, vol. 45, no. 8, 2017.

[17] K. I. Oyama, "A systematic investigation of several phenomena associated with contaminated langmuir probes," *Planetary and Space Science*, vol. 24, pp. 183–190, 1976.

[18] L. H. Brace, "Langmuir probe measurements in the ionosphere, measurement techniques in space plasmas," *Particles*, pp. 23–35, 1998.

[19] M. H. Piel, A. and C. T. Steigies, "Plasma diagnostics with langmuir probes in the equatorial ionosphere: I. the influence of surface contamination," *Journal of Physics D: Applied Physics*, vol. 34, p. 2643–2649, 2001.

[20] K. Hirao and K. Oyama, "Inaccuracies in electron density estimates due to surface contamination of langmuir probes," *Planetry and Space Science*, vol. 24, pp. 87–89, 1976.

[21] A. Barjatya, "Langmuir probe measurements in the ionosphere," Ph.D. dissertation, Utah State University, 2007. [Online]. Available: https://digitalcommons.usu.edu/etd/274

[22] W. E. A. et al, "Self cleaning langmuir probe," *Review of Scientific Instruments*, vol. 64, pp. 1253–1256, 1993.

[23] ——, "Contamination free sounding rocket langmuir probe," *Review of Scientific Instruments*, vol. 72, pp. 2052–2057, 2001.

[24] C. T. S. Hirt, M. and A. Piel, "Plasma diagnostics with langmuir probes in the equatorial ionosphere: Ii. evaluation of deos flight f06," *Journal of Physics D: Applied Physics*, vol. 34, p. 2650–2657, 2001.

[25] C. T. S. et al, "Surface property effects on langmuir probes launched on sounding rockets," *Proceedings of the 17th ESA Symposium on European Rocket and Balloon Programmes and Related Research*, 2005.

[26] A. B. et al, "Data analysis of the floating potential measurement unit aboard the international space station," *Review of Scientific Instruments*, vol. 80, 2009.

[27] ——, "Elevated electron temperatures around twin sporadic e layers at low latitude: Observations and the case for a plausible link to currents parallel to the geomagnetic field," *J. Geophys. Res. Space Physics*, vol. 118, pp. 7316–7328, 2013.

[28] A. KRUKOWSKI and I. KALE, "Simulink/matlab-to-vhdl route for full-custom/fpga rapid prototyping of dsp algorithms," *MATLAB DSP Conference*, 1999.

[29] V. Y. Sarge, "Evaluating simulink hdl coder as a framework for flexible and modular hardware description," Master's thesis, Massachusetts Institute of Technology, 2018. [Online]. Available: https://dspace.mit.edu/bitstream/handle/1721.1/119717/1078637048-MIT.pdf?sequence=1

[30] E. Pongratz and R. Cherian, "Performance evaluation of mathworks hdl coder as a vendor independent dfe generation," Master's thesis, Lund University, 2019. [Online]. Available: https://lup.lub.lu.se/luur/download?func=downloadFile&recordOId=8993578&fileOId=8993579

[31] L. Caseiro, D. Caires, and A. Mendes, "Prototyping power electronics systems with zynq-based boards using matlab/simulink–a complete methodology," *Electronics*, vol. 11, no. 7, 2022. [Online]. Available: https://www.mdpi.com/2079-9292/11/7/1130

[32] L. H. K. et al, "The scintillation prediction observations research task (sport) mission: Science conops," *American Geophysical Union*, 2020.

APPENDICES

APPENDIX A

Packet Structure

A.1 Generic Telemetry Packet Structure

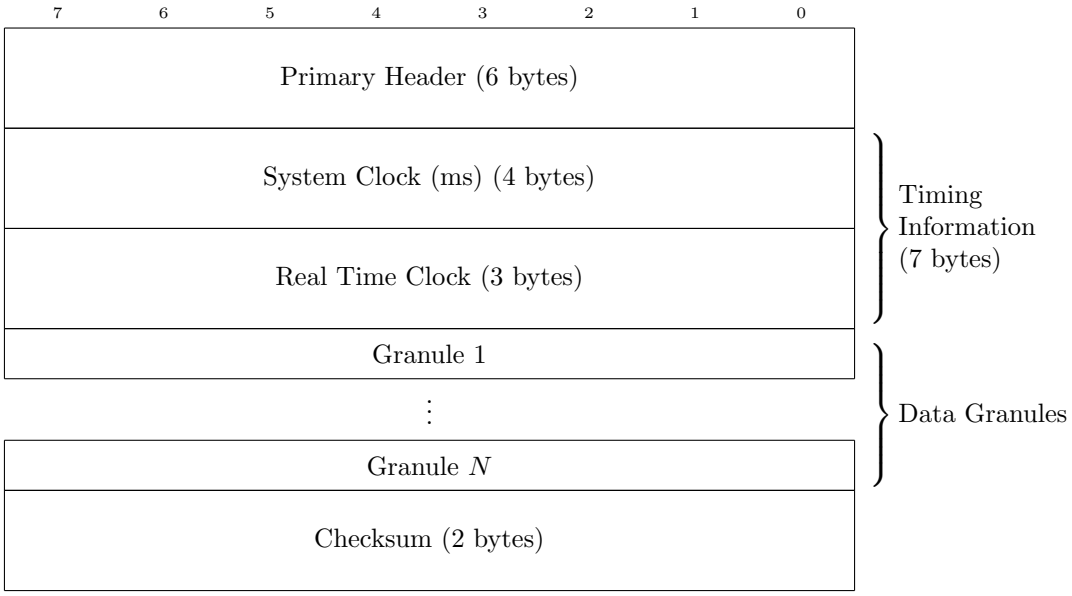

Fig. A.1: Generic Telemetry Packet

A.2 Floating Potential Probe (FPP)

A.1 FPP1&2 Packet and FPP3&4 Packet

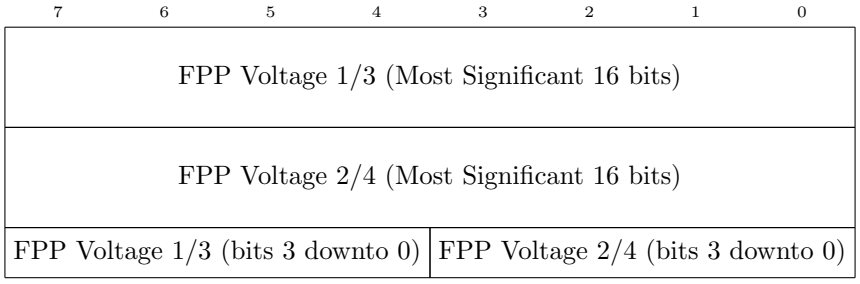

Fig. A.2: FPP Granule

A.2 EFW Packet

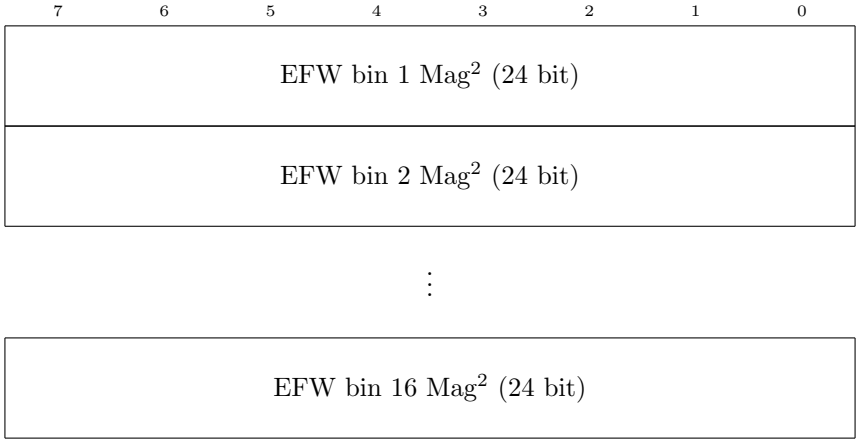

Fig. A.3: EFW Granule

A.3 Fixed Bias Langmuir Probe (FLP)

A.1 FLP

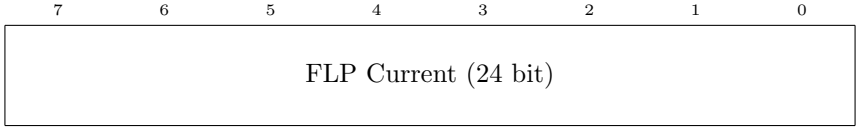

Fig. A.4: FLP Granule

A.2 FLW

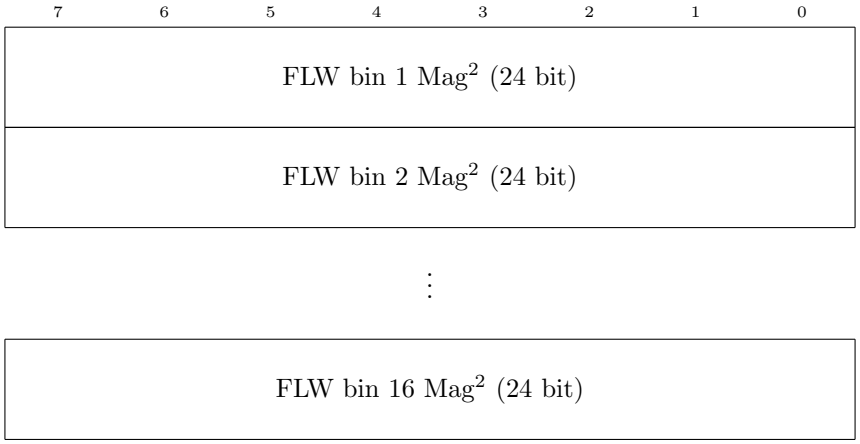

Fig. A.5: FLW Granule

A.4 Sweeping Langmuir Probe (SLP)

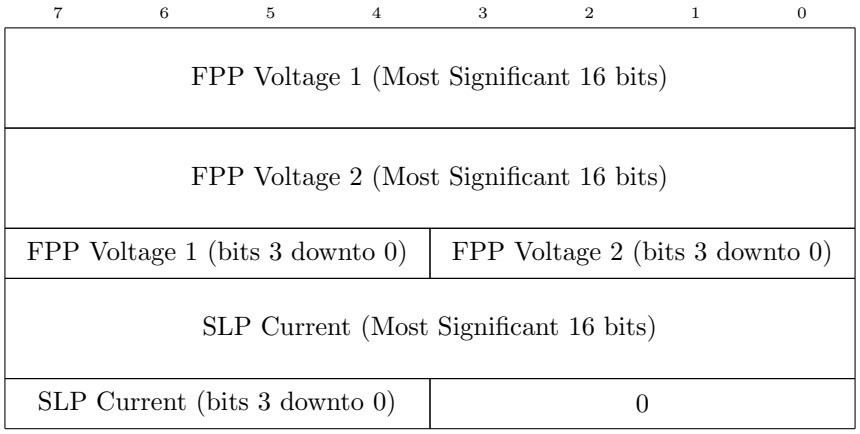

Fig. A.6: SLP Granule

A.5 Impedance Probe (IP)

A.1 TIP

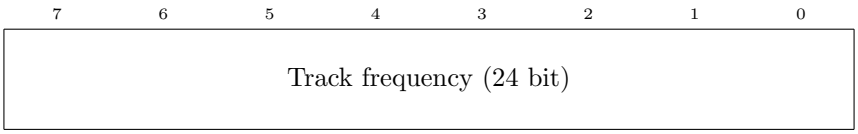

Fig. A.7: TIP Granule

A.2 QIP

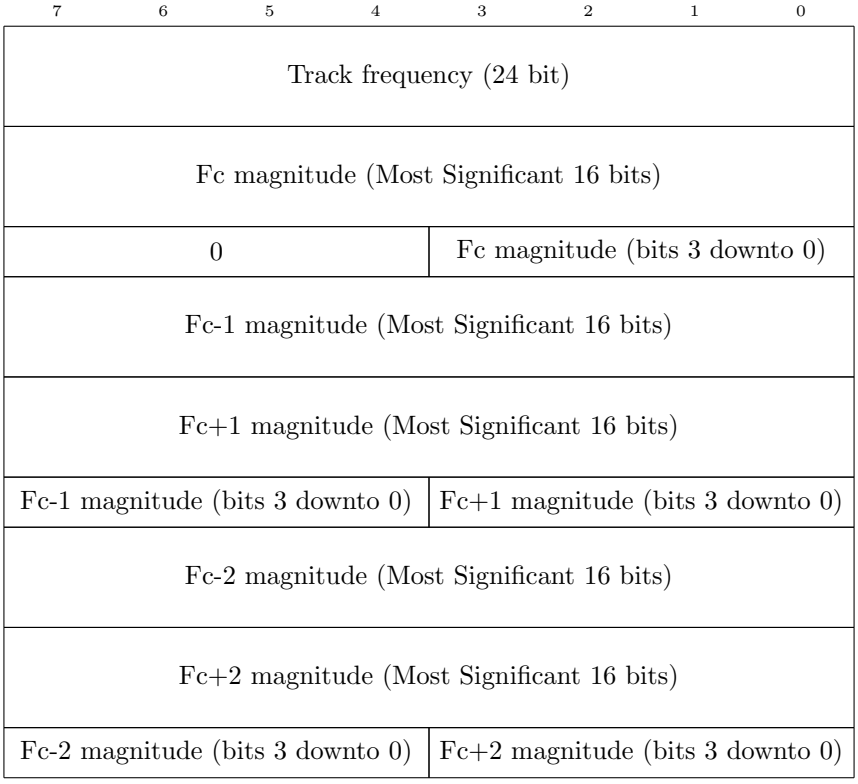

Fig. A.8: QIP Granule

A.3 SIP and LIP

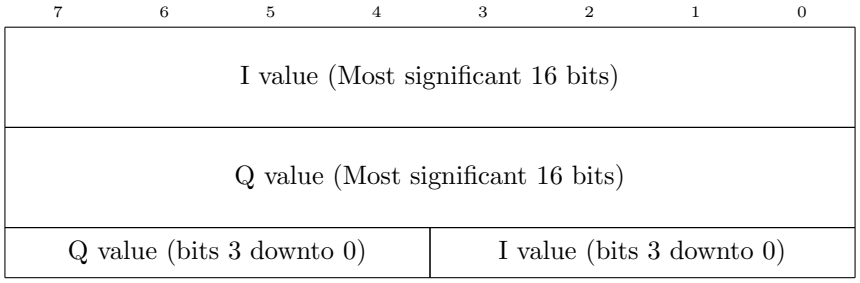

Fig. A.9: SIP and LIP Granule

APPENDIX B

Software Package Description

B.1 Overview

The thesis includes a software package. The software package includes the Simulink models designed in the thesis, the MATLAB scripts that configure the Simulink models, and the MATLAB scripts that analyze the simulation results. The software package includes four subfolders: model library, scripts, simulation results, and top models testbench. The file directory is shown in figure B.1.

B.2 model_library

The model library folder contains all of the Simulink models and subsystems used in the digital design for SWP2. The subfolders in the model library folder contain models for the FPP, EFW, FLP, etc.

B.3 scripts

The scripts folder contains the MATALB scripts that setup and configures the Simulink models. The project_setup.m script sets up the MATLAB environment and runs all of the other model configuration scripts that configure the Simulink models and subsystems. The scripts folder also contians the scripts that setup the simulation parameters used in this thesis.

B.4 simulation_results

The simulation results folder contains analysis scripts that produced the results in this thesis. The simulation results folder also contains the figures shown in 5.

B.5 top_models_testbench

The top models testbench folder contains the top models for testing each of the probes. The top models testbench folder also contains the testbench models for each of the probes in SWP2.

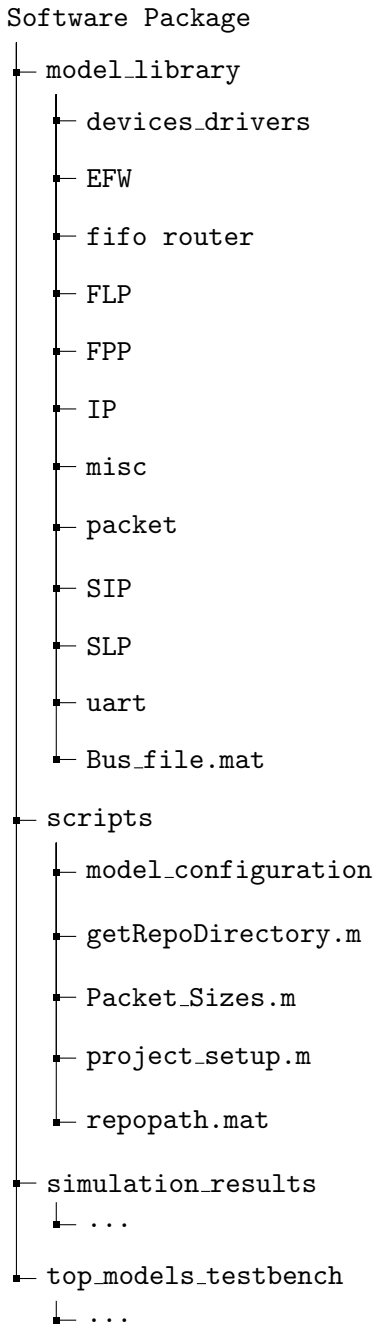

Fig. B.1: Software Package File Tree