

EE Design 2 Design Report
Title: Autonomous Agricultural Drone Station
Team: Drone Maestros

Contact Information:

Dieter Steinhauser
dsteinhauser@ufl.edu
954-774-9034

Ryan Laur
Ryan.laur.@ufl.edu
786-715-9718

Faculty Advisor Information:

David Arnold
darnold@ufl.edu

Trevor Free
tbrian.free@ufl.edu

Table of Contents

I.	Introduction.....	5
A.	Project Definition.....	5
B.	Scope of Project	5
C.	Product Expectations and Design Requirements	5
D.	Station Management Overview.....	6
II.	Background Information	6
A.	Beelink Overview - Ryan	6
B.	Raspberry Pico Overview - Dieter.....	6
C.	Capacitor Decoupling - Dieter.....	7
D.	ADC Theory - Dieter.....	7
E.	I2C Theory - Dieter.....	7
F.	Difference Amplifier - Dieter	8
G.	H-Bridge Circuit - Dieter.....	8
H.	UART Theory - Ryan.....	8
I.	JUCE Framework - Ryan	9
III.	Methods	9
A.	Local Network Configuration – Ryan & Dieter	9
B.	Internet Service Configuration – Ryan & Dieter	9
C.	ESP32 Wireless Drone Camera and Web Server - Ryan	9
D.	PC COM Device Driver – Ryan	10
E.	Pico UART Driver – Ryan & Dieter.....	11
F.	PC GUI Weather System – Ryan	11
G.	PC GUI Measurement Table.....	11
H.	PC GUI Window (Main Component)– Ryan	12
I.	Drone Flight State Machine - Ryan	12
J.	Power Chain Overview - Dieter.....	13
K.	Power Consumption Overview - Dieter	13
L.	ADC Configuration - Dieter	14
M.	Voltage References - Dieter.....	15
N.	ADC Channel Expansion - Dieter.....	15
O.	Reverse Polarity Protection - Dieter.....	16
P.	Output Power Switch - Dieter.....	17
Q.	Overvoltage Protection - Dieter	17
R.	Voltage and Current Measurement - Dieter.....	17
S.	Battery Charging Circuit - Dieter.....	19
T.	Battery Monitoring – Dieter	19
U.	BQ24650EVM MPPT Charge Controller – Dieter	19
V.	TPS1297AEVM DC/DC Buck Converter - Dieter	20
W.	I2C TPS55288EVM DC/DC Converter and Driver - Dieter.....	20
X.	TCA9548 I2C Switch - Dieter	20
Y.	DRV8871 H-Bridge Driver - Dieter.....	21
Z.	Drone Charging - Dieter	21
IV.	Results	22
A.	PCB Assembly - Dieter	22
B.	ADC Configuration - Dieter	23
C.	Voltage References - Dieter.....	23
D.	ADC Channel Expansion - Dieter	23
E.	Reverse Polarity Protection - Dieter.....	23
F.	Overvoltage Protection - Dieter	24
G.	I2C TPS55288 DC/DC Converter Driver – Dieter	24
H.	Voltage and Current Measurements - Dieter	24
I.	Drone Charging - Dieter	24
J.	Agisoft Orthomap Processing - Ryan.....	25
K.	Control Station GUI output - Ryan	25
V.	Discussion.....	25

A.	Main Computer Software and Hardware.....	25
B.	Power Management board	25
C.	Future Additions and Implementations	26
1)	GPS/RTK systems	26
2)	Drone Flight Control.....	26
3)	Internet Services	26
VI.	Conclusion	26
VII.	Acknowledgement	27
VIII.	References.....	27
IX.	Appendix	28
A.	Schematic and PCB - Dieter	28
B.	Bill of Materials - Dieter	36
C.	GitHub Design and Code Repository.....	37
D.	Additional Figures.....	38
E.	Omitted Designs.....	39
1)	Negative Voltage Reference - Dieter	39

Table of Figures

<i>Figure 1: Simple Block Diagram of the Base Station.....</i>	5
<i>Figure 2: Block diagram of remote/cloud system with Beelink Mini-PC</i>	6
<i>Figure 3: Raspberry Pico Pinout [15].....</i>	6
<i>Figure 4: Decoupling capacitor equation for minimizing voltage ripple caused by a transient current. [12].</i>	7
<i>Figure 5: Quantization of an Analog to Digital Converter on a sine wave.....</i>	7
<i>Figure 6: ADC Formulas</i>	7
<i>Figure 7: I2C Bus [7]</i>	7
<i>Figure 8: Single Byte Write and Read transactions [20]......</i>	8
<i>Figure 9: Simple difference amplifier with matched resistors [6]</i>	8
<i>Figure 10: CMOS H-Bridge and Current Paths [5].</i>	8
<i>Figure 11: Discrete H-Bridge and driver solution [6].....</i>	8
<i>Figure 12: UART data frame.....</i>	9
<i>Figure 13: JUCE GUI Application.....</i>	9
<i>Figure 14: Diagram of Local Area Network</i>	9
<i>Figure 15: ESP32-CAM Pinout and Features.</i>	10
<i>Figure 16: ESP32 Serial Port Printing IP Address.....</i>	10
<i>Figure 17: ESP32 Server Flow Diagram.....</i>	10
<i>Figure 18: Collaboration diagram for ComDevice.....</i>	10
<i>Figure 19: Call graph for run() thread handling.</i>	11
<i>Figure 20: Call/Caller graph for Buffer Handling.....</i>	11
<i>Figure 21: GUI display for current weather information.....</i>	11
<i>Figure 22: Call graph for fetchWeatherData()......</i>	11
<i>Figure 23: Call graph for changeListener callback.....</i>	11
<i>Figure 24: Collaboration Diagram of the Main Component.....</i>	12
<i>Figure 25: Call graph for default constructor.</i>	12
<i>Figure 26: Code flow diagram for flight.....</i>	12
<i>Figure 27: Call graph for begin flight function.....</i>	12
<i>Figure 28: Call graph for change Listener callback.....</i>	13
<i>Figure 29: Power chain block diagram.</i>	13
<i>Figure 30: Power Consumption by device</i>	14
<i>Figure 31: Current and power statistics for operating modes.....</i>	14
<i>Figure 32: Daily power generation and expenditure for operating modes</i>	14
<i>Figure 33: Pico 3.3V Noise Measurement (20MHz BW limit, 10x probe scaling, and ground spring attached).....</i>	14
<i>Figure 34: Pico ADC Connection [15].....</i>	14
<i>Figure 35: ADC LM4040 3V Reference circuit.....</i>	15

<i>Figure 36: Voltage Reference Schematic.</i>	15
<i>Figure 37: ADC Channel multiplexing circuit.</i>	15
<i>Figure 38: ADC multiplexing schematic.</i>	16
<i>Figure 39: Simple reverse polarity protection circuit. Vin is limited to the Vgs of the PMOS.</i>	16
<i>Figure 40: Reverse Polarity Protection circuit.</i>	16
<i>Figure 41: Simple PMOS power switch circuit. The PMOS is conductive when the S1 switch is open.</i>	17
<i>Figure 42: Active Power Zener for Voltage Clamping [6].</i>	17
<i>Figure 43: Example Voltage Clamping Circuit on 12V rail.</i>	17
<i>Figure 44: High and Low side current sensing schemes [4].</i>	18
<i>Figure 45: Unidirectional High Side Current Sensing with INA181 and implemented circuit [9].</i>	18
<i>Figure 46: Battery Charging Circuit.</i>	18
<i>Figure 47: Bidirectional High Side Current Sensing with INA181 for battery charging and discharging.</i>	19
<i>Figure 48: Coulomb Counting Algorithm [13].</i>	19
<i>Figure 49: BQ24650 Operating Conditions [2].</i>	19
<i>Figure 50: BQ24650EVM Equations [2].</i>	20
<i>Figure 51: TPS55288 Device Driver Register Mapping code structure.</i>	20
<i>Figure 52: I2C device and specific Driver Code structures</i>	20
<i>Figure 53: DRV8871 H-Bridge Breakout Board.</i>	21
<i>Figure 54: DRV8871 control logic and current paths [5].</i>	21
<i>Figure 55: Drone station landing pad.</i>	21
<i>Figure 56: Drone station landing quadrants and H-bridge.</i>	21
<i>Figure 57: Battery Polarity Detection Circuit</i>	22
<i>Figure 58: Threshold setting circuitry for the polarity detection circuit.</i>	22
<i>Figure 59: Pico 3V3 Noise Measurement (20MHz BW, 10x Probe and Scaling, Ground Spring attached).</i>	23
<i>Figure 60: LM4040 3V Noise Measurement (20MHz BW, 10x Probe and Scaling, Ground Spring attached).</i>	23
<i>Figure 61: ADC Channel Expansion Readings.</i>	23
<i>Figure 62: Reverse Polarity protection during polarity swapping event without a load.</i>	23
<i>Figure 63: Overvoltage protection clamping on 5V rail. (TIP142, 4.7V Zener, 22k resistor)</i>	24
<i>Figure 64: TPS55288EVM Test script iterating through various output voltages.</i>	24
<i>Figure 65: Agisoft Script that processes an orthomap from an available dataset.</i>	25
<i>Figure 66: Sparkfun GPS-RTK (GPS-22660) and Surveying Antenna (GPS-17751).</i>	26
<i>Figure 67: RFD900x-US telemetry Bundle.</i>	26
<i>Figure 68: Wiring Diagram of Pixhawk Flight Controller and RFD900x.</i>	26
<i>Figure 69: Power Board Schematic Page 1/6.</i>	28
<i>Figure 70: Power Board Schematic Page 2/6.</i>	29
<i>Figure 71: Power Board Schematic Page 3/6.</i>	30
<i>Figure 72: Power Board Schematic Page 4/6.</i>	31
<i>Figure 73: Power Board Schematic Page 5/6.</i>	32
<i>Figure 74: Power Board Schematic Page 6/6.</i>	33
<i>Figure 75: Reverse polarity protection schematic.</i>	34
<i>Figure 76: Power Board 3D top view.</i>	35
<i>Figure 77: Reverse polarity protection 3D top view.</i>	35
<i>Figure 78: Assembled station power board.</i>	38
<i>Figure 79: Battery Polarity Detection circuit in LTspice.</i>	38
<i>Figure 80: Negative Voltage Reference Circuit. Employs a Square wave generator, charge pump, and Vbe multiplier.</i>	39
<i>Figure 81: -5.08V Voltage Reference with 10mA load seen on channel 1. The charge pump and 10mA load required 21mA for operation.</i>	39
<i>Figure 82: Current supplied to the charge pump circuit.</i>	40

Autonomous Agricultural Drone Station

EEL3924C Senior Design Faculty Research Project with IoT4Ag
Team Drone Maestros

Dieter Steinhauer
Department of Electrical and Computer Engineering
University of Florida
Gainesville, Florida, USA
dsteinhauser@ufl.edu

Ryan Laur
Department of Electrical and Computer Engineering
University of Florida
Gainesville, Florida, USA
ryan.laur@ufl.edu

Trevor Free
Department of Electrical and Computer Engineering
University of Florida
Gainesville, Florida, USA
trevor.free@ufl.edu

David Arnold
Department of Electrical and Computer Engineering
University of Florida
Gainesville, Florida, USA
darnold@ufl.edu

Abstract— The Autonomous drone port is a collaborative research project with the Agribugs company and Internet of Things for Precision Agriculture (IoT4Ag). Our purpose is to design and build a system that serves as an autonomous drone port for deployment in agricultural environments. The goal is a fully autonomous station requiring minimal human intervention. Currently, the project has several individually working electrical systems that require an operator to power, set up, and collect data. System integration of the solar charging, power management, environment sensors, data collection, and a web server are desired for the project. The Station should provide a hub for drone charging and data upload to cloud storage. Duplex communication should exist between a remote-control GUI and the drone station.

I. INTRODUCTION

The autonomous agricultural drone station will serve as a landing base, hangar, solar energy collector, recharging platform, data collector, and communication hub for a quadcopter drone. Agribugs and IoT4Ag are creating an autonomous drone system to gather agricultural data for research interpretation of crop health. Agribugs currently uses a series of off-the-shelf systems and human operators to gather the desired agricultural data. Complications of isolated systems, skilled human operation, and weather conditions limit the quantity and quality of data collected. Working towards a unified station that provides drone shelter, charging, data interpretation, and remote user interaction will allow for widespread deployment of agricultural inspection drones and mitigates the previously stated complications.

A. Project Definition

Our Project is to work towards an autonomous drone station that is capable of deployment in remote agricultural locations. Integration of various off the shelf parts and existing systems into a unified station is necessary for this project. A primary task of the station is to provide a landing zone and shelter for the autonomous drone. The station should be a central communication point between the autonomous drone and the remote user. Communications between the drone and the station consist of GPS/RTK, camera, and flight controls data. The station will have an internet connection for remote access of media and systems. Solar charging should be utilized for long-

term field operation. The station would also charge the drone without user intervention.

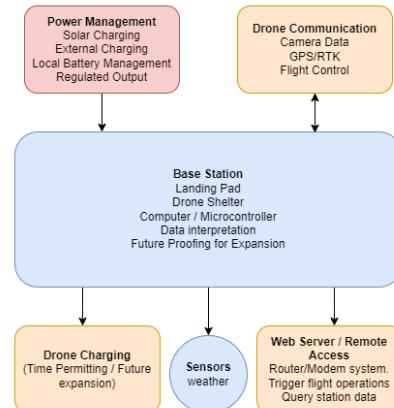


Figure 1: Simple Block Diagram of the Base Station

B. Scope of Project

The scope of our contribution shifted through the semester since our team is taking the first steps towards system integration. The team will lay the foundation for the drone port systems integration and document systems thoroughly for future development. The project should be unifying existing systems that Agribugs has in place, adding off-the-shelf systems to supplement our designed circuits. Our focus lies on creating a foundation for the project by creating a robust power system and computer interface for expansion of systems in the future. Primary goals lie in smart power management and remote access to drone station systems.

C. Product Expectations and Design Requirements

Realizing this project will require the following tasks:

- Implement solar and AC/DC battery charging.
- Develop power management and protection circuitry while remaining efficient.
- Sense voltage and current of various power rails to determine energy statistics.
- Create a web server GUI for interaction with the station.

- Integrate systems with existing Agribugs drone landing station.
- Communicate with the drone for flight control and mission planning.

The final product should incorporate the following design criteria:

- Performance
 - o Light weight design
 - o Temperature Tolerance
 - o Multi-month operation without intervention
 - o Low latency media transmission
- Serviceability
 - o Sturdy and dependable design
 - o Fault tolerance
 - o Easy repairability
- Features
 - o Internet connection
 - o GPS/RTK link
 - o Web Server
 - o Solar Charging
 - o Power analytics
 - o Camera data capture/processing
 - o Expansion capabilities
- Reliability
 - o Stable operation
 - o Weatherproof chassis
 - o Long service life
- Cost
 - o Off-the-shelf parts
 - o Replaceable or repairable parts
 - o Ideally low cost
- Safety and Security
 - o Electrically protected.
 - o Secure remote connection

D. Station Management Overview

The main computer will initiate data transfer from the drone to the media server, log mission critical data, manage weather data, and make camera data available to the user. Once the drone lands back at the station, a Wi-Fi data dump will occur, and the main computer will encode data into a remote viewing format and make them available on the media server. The camera data from the drone will be used to create an Ortho-map that will be used by researchers and farmers to analyze crop health.

II. BACKGROUND INFORMATION

A. Beelink Overview - Ryan

The Beelink mini-PC has four USB ports – 3 USB3.2 ports, and 1 USB 2.0 port. These ports will be used to communicate with the other devices. Other features include Wi-Fi 6, Bluetooth 5.2, ethernet, HDMI, and a 500GB SSD. The beelink PC can run Windows or Linux operating systems. Power consumption on the Beelink idles around 10W, but maximum power draw is documented to be 60W.

B. Raspberry Pico Overview - Dieter

The Raspberry Pico is a development board with an RP2040 microcontroller chip. It has 264KB of SRAM and 2MB of flash

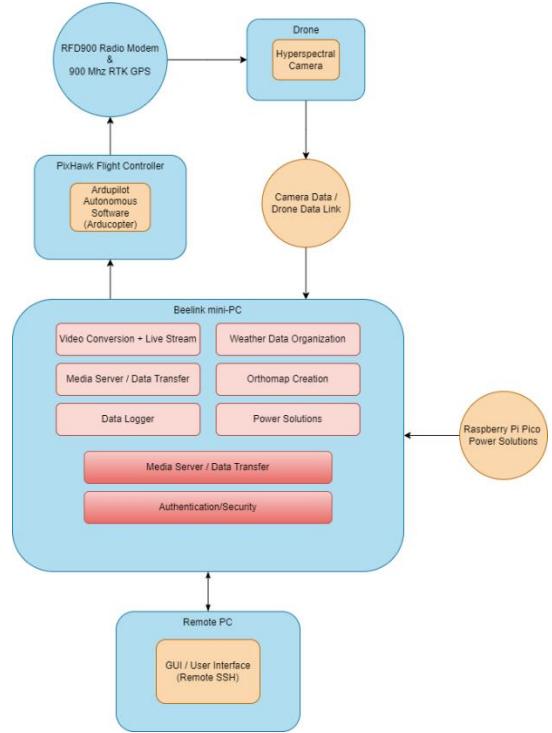


Figure 2: Block diagram of remote/cloud system with Beelink Mini-PC

memory. Development for the Pico is versatile as the platform capable of being programmed in MicroPython, CPython, C, and Rust. The Pico has a dual core processor with a warrantied clock speed of 133MHz, set to 125MHz out of the box. The Pico is easily overclocked to over double the stock processor speed, with simple overclocking available to 270MHz. There are 40 GPIO pins on the board, with SPI, I2C, and UART communication capability. The Pico comes equipped with a 12-bit 500ksps ADC with 5 channels total. One channel is configured to an RP2040 internal temperature sensor, and three are tied to GPIO pins on the Pico. The Pico also has a timer with four alarms, a Real-Time-Counter (RTC), and sixteen Pulse-Width Modulation (PWM) channels. The Pico also comes in a Wi-Fi and Bluetooth enabled version called the Pico W. The Pico microcontroller series are inexpensive, starting at \$4 and are great low power devices [15].

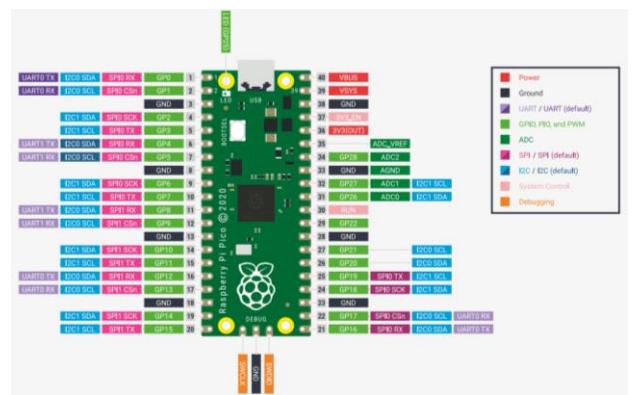


Figure 3: Raspberry Pico Pinout [15]

C. Capacitor Decoupling - Dieter

When a capacitor is placed from a DC node to ground, this is referred to as a decoupling or bypass capacitor. Decoupling and bypass capacitors have slightly different reasons for use but are practically synonymous. A bypass capacitor prioritizes removal of unwanted high frequency noise from a node on the circuit since it acts as low impedance (a short) for high frequencies. Bypass caps may also create an RC filter.

$$C = \frac{\Delta I}{\Delta V / \Delta t} \quad (3.4)$$

$$\text{For example, } \frac{20 \text{ mA}}{100 \text{ mV}/5 \text{ ns}} = 0.001 \mu\text{F} \text{ or } 1000 \text{ pF}$$

where ΔI = current transient

ΔV = allowable power supply voltage change (ripple)

Δt = switching time

Figure 4: Decoupling capacitor equation for minimizing voltage ripple caused by a transient current. [12].

Decoupling usually avoids AC ripple, noise, and overall voltage discontinuities that can occur in a voltage rail. Capacitors used for decoupling are generally next to a device or IC to clean up voltage used for power or reference. In the application of decoupling, a capacitor is added to remove voltage ripple caused by a transient current load. If you have a microprocessor that suddenly requires a surge in current, a decoupling capacitor will minimize voltage ripple based on the current required, time of the surge, and capacitance.

D. ADC Theory - Dieter

Analog to Digital Converter's are the primary method of capturing analog data using a microcontroller. ADC's capture analog signals such as sound or light waveforms and convert them to a digital signal for processing information. An ADC takes continuous magnitude data from a continuous time domain into discrete magnitude data in a discrete time domain [1]. The conversion quantizes the signal based on the resolution, reference voltages, and sample rate of the ADC, introducing small errors into the signal as noise.

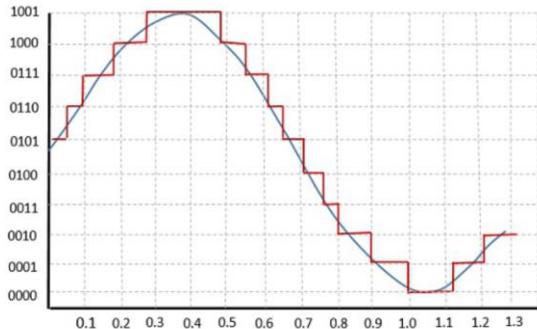


Figure 5: Quantization of an Analog to Digital Converter on a sine wave.

Resolution and reference voltages of an ADC produces quantization errors within the amplitude domain of a signal, where the sample rate of an ADC produces quantization errors within the time domain. Resolution gives the number of discrete values an ADC can produce, where the reference voltages

provide the span of voltages the ADC can receive as a signal. Ground is often defaulted to the lower reference voltage to simplify ADC implementation. As a result, the voltage resolution in volts per bit is given as the span between the reference voltages divided by the number of discrete values the ADC resolution can produce. The digital code made by the ADC multiplied by the voltage resolution of the ADC will return the voltage recorded by the ADC. Quantization produces rounding errors in the approximation of the analog voltage, which is worsened by noise and jitter that exists on the signal prior to the ADC conversion. Signal noise can significantly reduce the effective number of bits to which an ADC is accurate.

$$V_{res} = \frac{V_{ref_H} - V_{ref_L}}{2^{ADC_BIT_RES}} \quad V_{out} = ADC_{code} * V_{res}$$

Figure 6: ADC Formulas

Sample rate of an ADC produces quantization errors within the time domain of the signal. The sampling rate of an ADC should ideally be greater than twice the highest frequency being recorded, otherwise aliasing will occur. Aliasing can also be mitigated by adding a low-pass filter to the ADC input, removing frequencies above half of the sampling frequency. Oversampling is also commonly employed as it can reduce noise and improve bit-depth.

E. I2C Theory - Dieter

I2C is a half-duplex serial communication protocol created by Phillips Semiconductor in the 1980's for low-speed communication in embedded systems. [7] The I2C protocol consists of a serial data line (SDA) and a serial clock line (SCL). These serial wires receive weak pull-up resistors to default the communication wire to the logical high voltage. Raw communication speed is limited by the device interacting, however 100kbps or 400kbps is common. I2C is half-duplex as data is bidirectional but can only communicate in one direction at a given time.

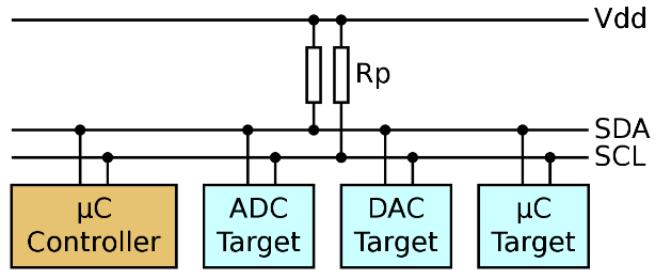


Figure 7: I2C Bus [7]

Interaction between I2C devices utilizes addresses, allowing for multiple master and multiple slave devices. This allows for circuit design simplicity as only two wires are required regardless of the number of devices. However, addressing of devices will fail if more than one device uses a specific address. I2C is also limited in communication speed because of the two-wire interface but is further restricted by device addressing overhead for each transaction.

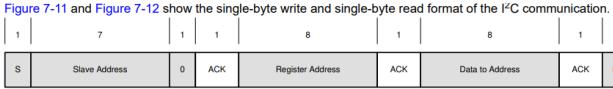


Figure 7-11. Single-byte Write

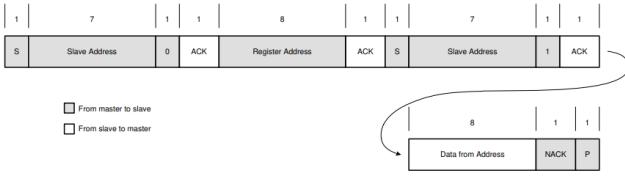


Figure 7-12. Single-byte Read

Figure 8: Single Byte Write and Read transactions [20].

Communication between I2C devices occurs when a master device produces a start signal to initiate communication. The master device will start providing a clock signal while a slave device is addressed over the data line with its unique 7-bit address. This is the address data frame. After every data frame, the master queries for acknowledgement from the slave device. The master device then commands a read or write to the slave device using a register data frame. In the case of writing data, a write data frame will then be sent to the slave device. In the case of reading data, another address data frame is sent before the slave device returns the data. Multi-read and multi-write transactions can occur where extra data frames are added to the end of transactions for communication of larger consecutive amounts of information.

F. Difference Amplifier - Dieter

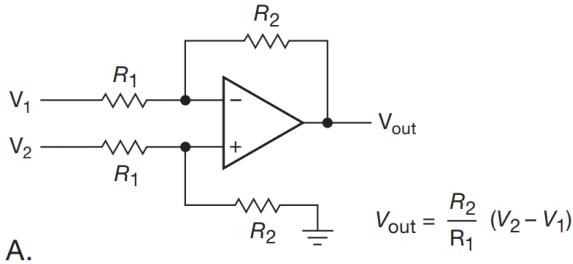


Figure 9: Simple difference amplifier with matched resistors [6]

Difference Amplifiers are simple way to find the difference between two voltages, also known as a voltage subtractor. An output voltage is created that is proportional to the difference between two voltages. If all resistors are the same value, then the circuit is considered a unity gain differential amplifier and the output voltage is equal to the difference between the input voltages. An important note of this circuit is that the output can be positive or negative with respect to the reference or ground. Depending on the circuit, a dual supply may be needed, or the reference point may need to be offset. Either of these approaches could be used to capture negative values of the voltage difference observed.

G. H-Bridge Circuit - Dieter

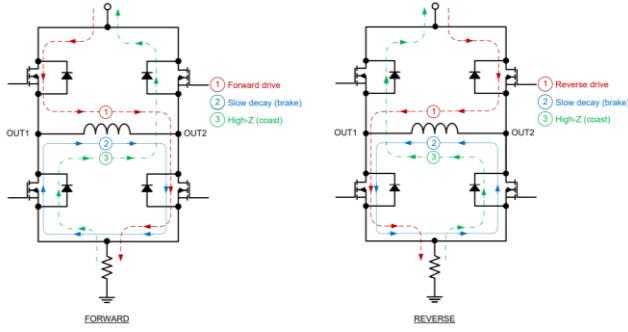


Figure 10: CMOS H-Bridge and Current Paths [5].

An H-bridge is a circuit configuration consisting of four transistors, primarily MOSFETs or BJTs configured in an H pattern. Each side of an H-bridge has a complementary set of transistors where only one transistor on each side is active at a given time. If two transistors on one side are on, a short circuit will occur as the attached transistor try and source and sink as much current as possible. During normal operation, current flows through an attached device in one of two directions.

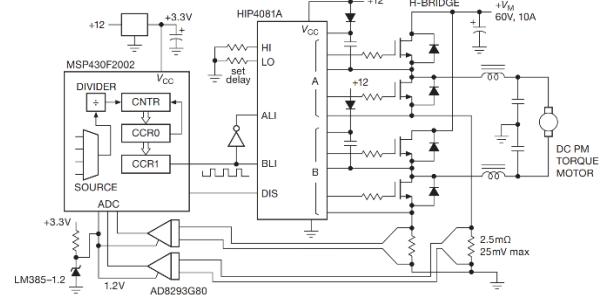


Figure 13.12. Controlling a torque motor with pulse-width modulation.

Figure 11: Discrete H-Bridge and driver solution [6]

The circuit essentially acts as an alternating set of switches so that the current flow or polarity may be swapped on a device. H-bridges are commonly used for motor drivers because of their ability to swap polarity, allowing forward and reverse motion control. Braking and coasting configurations are also available for abrupt or natural decaying stops respectively. H-Bridge circuits can be made discretely or be found in integrated packages often with driver circuitry included. Driver circuitry simplifies the process of controlling an H-bridge and insures safe operation. Often, H-bridge drivers only require static digital values or a PWM signal to change the average voltage and current applied.

H. UART Theory - Ryan

Universal Asynchronous Receiver/Transmitter (UART) is a hardware communication protocol used for serial communication. UART communicates data sequentially over a single channel, making it useful in long-distance or low-speed applications. UART operates asynchronously, so it doesn't use a shared clock signal for synchronization between the transmitter and receiver. Instead, the devices will use the same baud rate (bits per second) to communicate.

Start Bit (1 bit)	Data Frame (5 to 9 Data Bits)	Parity Bits (0 to 1 bit)	Stop Bits (1 to 2 bits)
------------------------	------------------------------------	-------------------------------	------------------------------

Figure 12: UART data frame.

Each data frame starts with a start bit, which triggers the receiver to start interpreting incoming bits. Following the start bit, a range of 5 to 9 data bits is sent to send the actual communication. There is an option parity bit used for error checking. This can be set to none, even, or odd. Finally, one or two stop bits are sent to signal an end of the data frame.

I. JUCE Framework - Ryan

JUCE Is an open-source C++ application framework, specifically used for its robust handling of complex user interfaces. In our project, JUCE serves as the backbone for developing the Graphical User Interface (GUI) for the control hub, and facilitates the efficient management of resources, threaded processes, and visual components. The goal is to handle multiple processes concurrently, without affecting the user experience. JUCE and C++ both provide a great set of tools to handle background tasks and user interactions seamlessly. This will ensure that the GUI and controls remain responsive while handling intensive tasks like server communication, and data transmission.

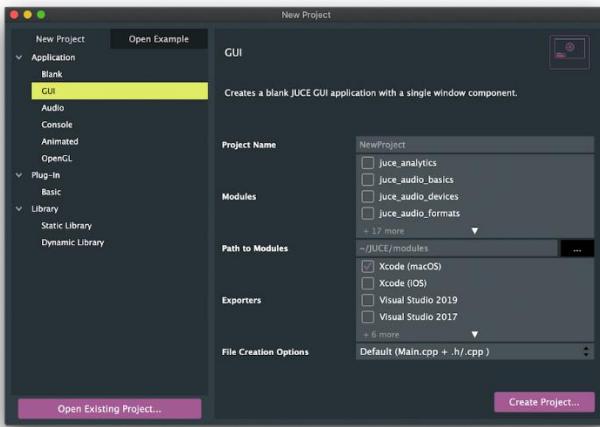


Figure 13: JUCE GUI Application

JUCE manages C++ projects using the Projucer, a project management tool that generates the framework necessary to build the application the user desires. In this case, a GUI application would be applicable to the Drone Station project. JUCE provides a comprehensive set of tools for customizing GUI components, which simplifies the development of a complex interface. JUCE contains drawing elements, menus, and event-driven listeners to provide control and feedback to the user. Drone hub operations are greatly simplified with JUCE providing the framework for the GUI components.

III. METHODS

A. Local Network Configuration – Ryan & Dieter

A TPlink AC750 Travel router was used to create the local network on the station. This device was chosen for its small form factor, low power consumption, and mobile features. If a wired modem connection to the router is present, the router will create a Wi-Fi network and produce an SSID. If a mobile hotspot is used, the SSID and password will be the same as the mobile hotspot. Either way, the travel router creates a robust local area network (LAN) for devices such as the PC, ESP32, raspberry Pico, and other wireless devices to connect.

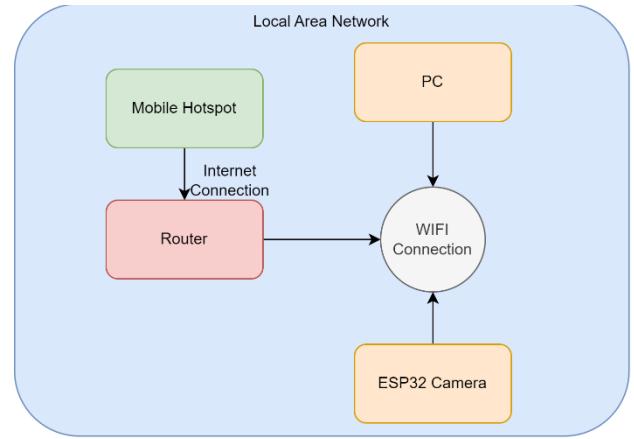


Figure 14: Diagram of Local Area Network.

B. Internet Service Configuration – Ryan & Dieter

To configure the internet service, a mobile hotspot was used in conjunction with a TP-Link AC750 wireless route. In the setup options, the WIFI extender mode was selected, and then the wireless 5GHz is set up with the desired hotspot. After this setting is enabled, the travel router acts as a range extender for the mobile hotspot, rebroadcasting the SSID of the mobile hotspot network. This effectively allows for access to the wide area network (WAN) or the internet while also having the benefits of a strong and dedicated LAN. For future implementations of the drone station using either a 4G-LTE modem or satellite internet modem for WAN/internet connection, the AC750 can be reconfigured to act as a router instead of a range extender. These services were not explored due to their hardware costs and monthly service fees. We felt that the addition of either hardware or services was not needed at this point in development and would waste funds.

C. ESP32 Wireless Drone Camera and Web Server - Ryan

The ESP32-CAM is a system from Ai-Thinker that is used to act as the microprocessor and camera. In a future setup, a higher quality camera system would be used that communicates directly with the drone; However, for demonstration purposes, an ESP32-CAM was used.

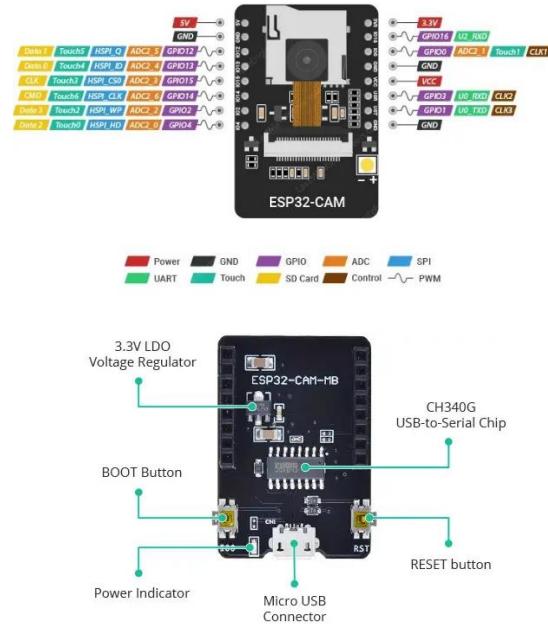


Figure 15: ESP32-CAM Pinout and Features.

The ESP32-CAM can be programmed using an FTDI adapter, but in this case the ESP32-CAM-MB adapter (programmer) was used. Setting up the device, the ESP32 Arduino boards must be installed, and the AI Thinker ESP32-CAM board is selected. Then, once the programmer was connected via USB, the corresponding COM port was selected, and the device is ready to be programmed.



Figure 16: ESP32 Serial Port Printing IP Address.

The ESP32-CAM takes pictures on a 2 second interval, and 10 total captures are saved for demonstration purposes. First the camera is initialized and is connected to WIFI. Note, the SSID and password of the Wi-Fi need to be specified in the code. Once connected to Wi-Fi, the Terminal window will display the IP address. This information is crucial for the GUI to connect to the device.

The ESP32 will create a server once connected to the internet and will maintain a “Photos Not Ready” message at the [IP Address] /done. HTTP requests are made from the GUI to the ESP32, and when a request is made at the /beginFlight address, the ESP32 will begin taking pictures. Once it is finished taking the pictures, it will read them from the SD card and update the /done address to “Serving Photos”. Once a photo is uploaded by sending an “image/jpeg” request to the /picture server address, the GUI will receive and download it. The root (/) address is used to send the total list of pictures to the GUI for maintaining an index of files. Once there are no more images to upload, the /done address will send the message “Serving Photos Done” when a request is received.

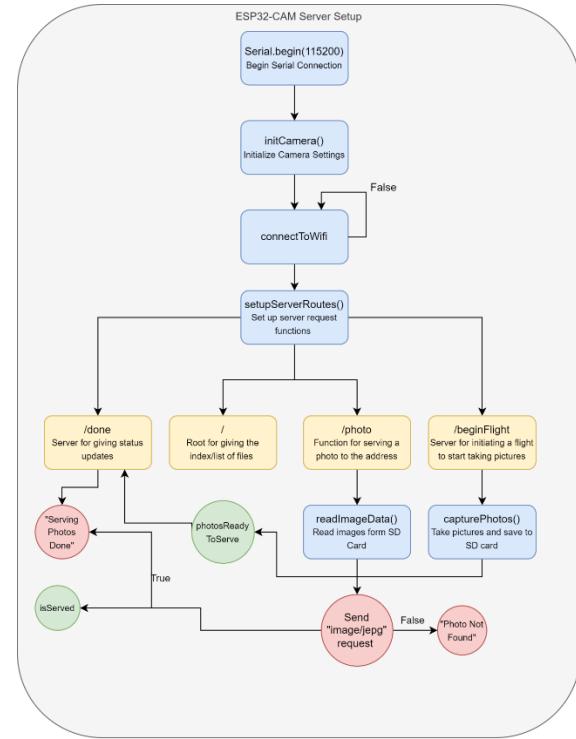


Figure 17: ESP32 Server Flow Diagram.

D. PC COM Device Driver – Ryan

The PC COM device driver comes from the ComDevice class in the control station GUI program. When the ComDevice object is created in the MainComponent class instantiation,, a new thread is created that has ownership of the COM process. An enum is selected telling the run() method of the thread which instructions to execute. First the serial port is opened, and proceeds to try to open a serial port for a maximum of three attempts.

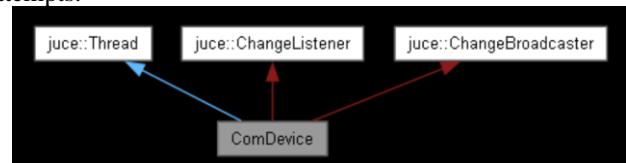


Figure 18: Collaboration diagram for ComDevice.

The ComDevice uses a custom module called juce_serialport to read COM paths from the PC. If the connected path is found, it will update and configure the selected serial port. The baud rate is set at 175 kHz, 8 data bits, even parity, and two stop bits. The ComDevice is setup to both read and write to via serial, but the current implementation only reads from the Raspberry Pi Pico. If writing is necessary, the user can update the eThreadTaskWriteSerialPort section of the run() method for their own purposes.

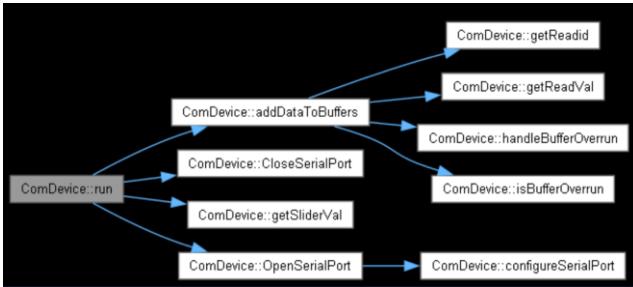


Figure 19: Call graph for run() thread handling.

When reading from the serial port, it reads 3 bytes of data at a time using the `SerialPortInputStream` object. The first byte consists of the ID (`readData[0]`), and the next two bytes are the data. To combine the bytes for the data, `readData[1]` is multiplied by 256 to get the MSB, and added to `readData[2]` to combined the LSB. The final value is divided by 1000, because the Raspberry Pi Pico sends the data multiplied by 1000. This allows the Pico to send UART data as unsigned numbers greater than zero, and the gives several decimals of accuracy.

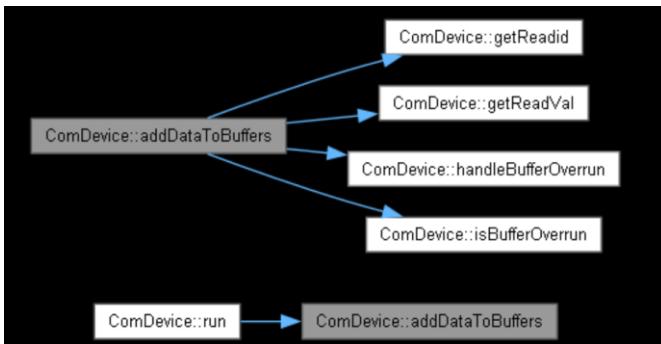


Figure 20: Call/Caller graph for Buffer Handling.

The IDs and values are stored to a data buffer to prevent reading data faster than the GUI can update it. A `ChangeListener` and `ChangeBroadcaster` is used to tell the `MainComponent` GUI that there has been a change on the data buffer after a read has occurred. Appropriate methods for handling the ID and value Queue data structures were implemented, handling buffer overrun/underruns and logging when appropriate.

E. Pico UART Driver – Ryan & Dieter

The Pico UART driver makes use of the Raspberry Pico's hardware UART system to transmit data to the main PC for GUI interpretation. Over UART an ID is transmitted followed by an unsigned 8-bit integer for interpretation on the PC. This allows for the PC to interpret what datapoint it is receiving and interpret it accordingly. A separate function iterates through all the available ADC values and transmits each datapoint with its identifier code. The ID value is 1 byte, and is combined with the value (data) which is two bytes, in Big-Endian format. Between each transmission of 3 bytes, the Pico waits for 20ms to allow prevent overloading the UART buffer. If the ID sent is either a 69 or a 70, the function first divides by 100, and then multiplies by a 100 in the actual sending of the serial data. This is because all of the other values are multiplied by 1000 to provide proper accuracy/numbers greater than 0, but the IDs of 69 and 70

correspond to values that are much smaller, and only need to be multiplied by 10.

F. PC GUI Weather System – Ryan

Weather information is updated from a simple website (<https://wttr.in/>) that provides weather data for the current GPS location in JSON format. The `fetchWeatherData()` function will first check if the internet is connected, and then create a new thread to parse the JSON file, evaluate the current conditions, and update the display.

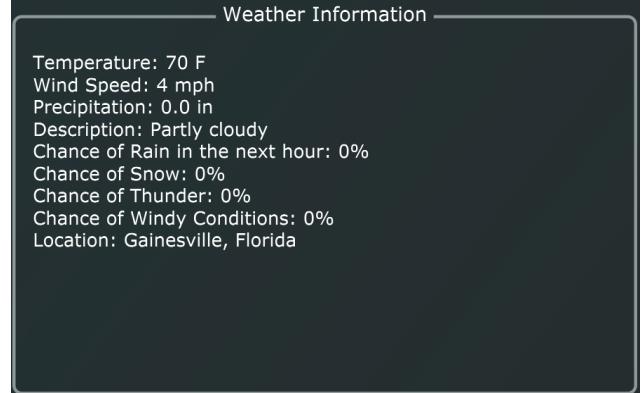


Figure 21: GUI display for current weather information.

The GUI weather display shows the current temperature, wind speed, precipitation, current conditions, and information on if there is going to be adverse weather conditions in the next hour. All GUI operations are performed asynchronously outside of the weather thread, in case there is a timeout when requesting data from the website. Fetching the weather data occurs on a timer, which varies depending on the battery state. The weather will not update if the battery is on critical mode, and will update much faster if the battery is charging/in performance mode.

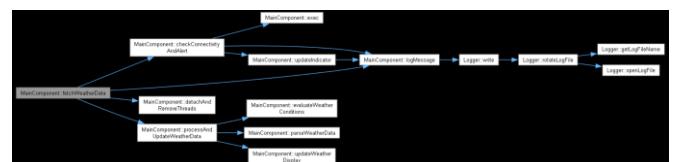


Figure 22: Call graph for `fetchWeatherData()`.

G. PC GUI Measurement Table

The GUI measurement table is where all the power system data is displayed. The GUI waits for serial data to be received, and then updates the GUI accordingly. A buffer is used to store the COM serial data, which contains an ID, and a corresponding value. The ID pertains to component to be updated, and the values are what should be displayed.

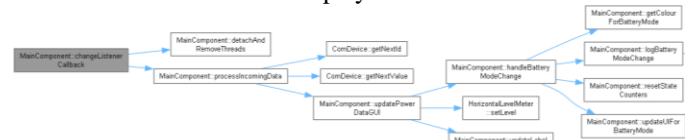


Figure 23: Call graph for `changeListener` callback

This method processes incoming data on a new thread when the listener callback function is triggered. The listener is waiting

for data from the ComDevice object, and it retrieves the next ID its associated value from a buffer. It then iterates through a batch of power data and updates the GUI components, accordingly, based on the ID (component), and value (voltage or current). The labels are only updated if changed.

H. PC GUI Window (Main Component) – Ryan

The main GUI window lies in the MainComponent class of the control station program. It applies ChangeBroadcasters and ChangeListeners to effectively “listen” for COM data from the Raspberry Pi Pico and runs multiple timers for the various states. MainComponent draws the drop-down window “ComboBox” to display the various battery states, draws horizontal level meters to display the current battery health and percentage, and updates various states and information using timers. All server requests and non-GUI related functions use threads and asynchronous updates so the program runs without hangs and interruptions.

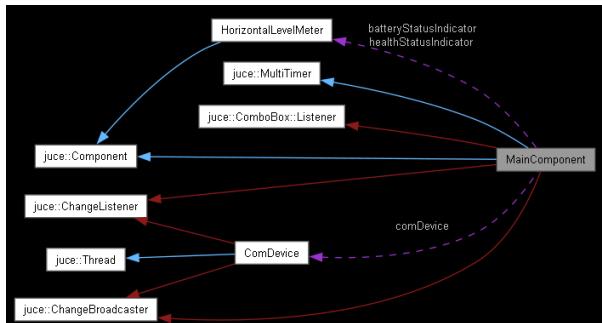


Figure 24: Collaboration Diagram of the Main Component.

The MainComponent default constructor setups up various indicators and labels for COM Port status, internet status, ESP32 status, and battery state. It configures the shapes and texts for these indicators, and the labels to reflect the status of each component. Default initializations include iterating through the voltage and current default values, setting the text and bounds of the GUI components, and making them visible within the window. Based on the index of the iteration, it assigns specific names to the voltage and current labels to be displayed to the user.

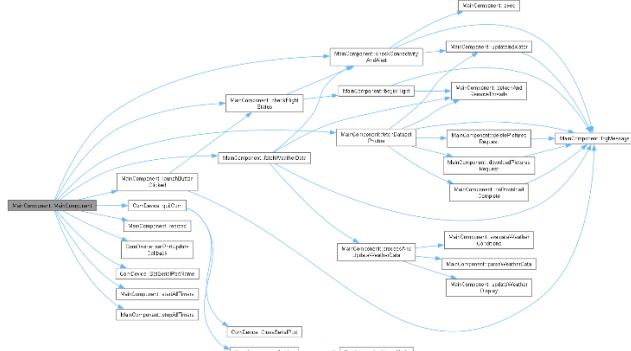


Figure 25: Call graph for default constructor.

I. Drone Flight State Machine - Ryan

Several states and flags manage the logic for determining whether a drone is ready to fly, which will signal pictures to be taken on the ESP32. The decision comes from several hardware

connectivity checks, weather data validation, and timer-based functions. The state machine flowchart below depicts the implementation through several functions within the MainComponent class of the control station.

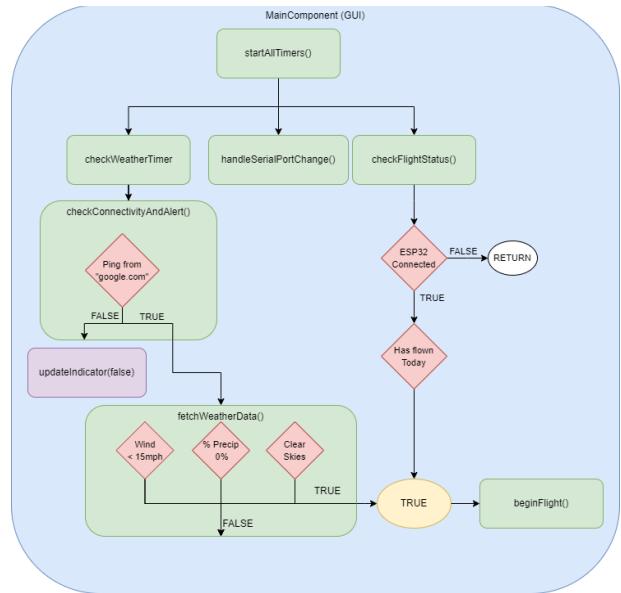


Figure 26: Code flow diagram for flight.

First, the program checks internet connectivity by attempting to ping an external server (google.com). If the ping request succeeds, it will then fetch and process weather data from wttr.in, with failed attempts managed in a separate threaded process. During the weather check, it determines if there are Sunny, Clear, Cloudy, or Overcast skies, with clear skies and wind speeds less than 15mph. If there is no precipitation forecast for the next hour, then it proceeds. On a separate parallel process, the program evaluates the connection status of the ESP32 camera and the Raspberry Pi Pico, and whether the drone has already completed a flight for the day. This evaluation is necessary to prevent repeated operations and ensure the drone's operations are limited to the operational time window between 8 am and 4 pm.

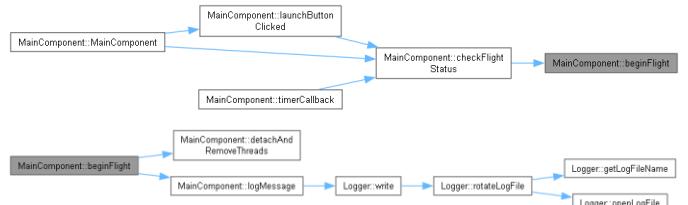


Figure 27: Call graph for begin flight function.

While the flight function being executed on a separate thread, the main thread will asynchronously log the status of flight to the UI window, and a log file located in the program directory. Threads are cleaned up on a timer and joined appropriately to save resources.

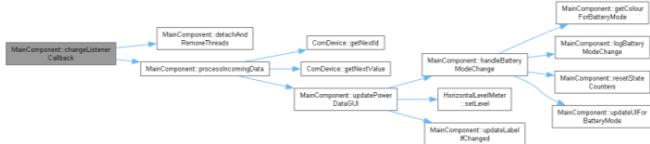


Figure 28: Call graph for change Listener callback

This method processes incoming data on a new thread when the listener callback function is triggered. The listener is waiting for data from the `ComDevice` object, and it retrieves the next ID its associated value from a buffer. It then iterates through a batch of power data and updates the GUI components accordingly, based on the ID (component), and value (voltage or current). The labels are only updated if changed.

J. Power Chain Overview - Dieter

The Autonomous drone station requires a robust power management system for long term field operation and future expansion. Our project team was provided with a 30W solar panel and charge controller for solar charging of the station. Realistically, a larger panel configuration will be necessary for operation of the system. A 12V Lithium Iron Phosphate battery bank will be used to store and supply energy to the station. Voltage and current measurements will occur at the MPPT solar controller and battery to observe charging and discharging. The various electronics on the station require a selection of DC/DC buck-boost power rails for continuous and efficient operation. A 20V/5A rail will be implemented for drone charging support. A buck-boost converter will also be used for a 20V/5A rail to provide power to the onboard computer. Smaller devices such as motors, sensors, microcontrollers, will rely on 12V/5A and 5V/5A rails respectively. Each DC-DC rail will have power protection circuitry for overcurrent and overvoltage conditions. Finally, all output rails will be measured and filtered before supplying the station with power. Measurements will be read into a Raspberry Pico microcontroller for interpretation and send collected data to the main computer.

For better design integration, we chose to incorporate a series of Texas Instruments (TI) evaluation boards for both the MPPT charge controller and DC/DC converters. These devices were chosen for their ease of access and documentation, allowing future development of a fully integrated power supply PCB. Efficiency of devices is 95% or higher at load. Current and voltage limiting were also key features of these devices. Implementation of sensing circuitry will be required.

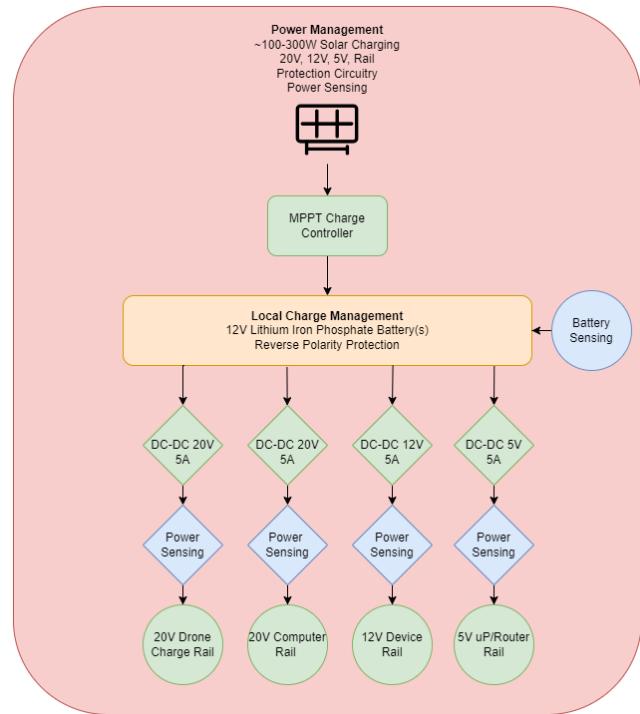


Figure 29: Power chain block diagram.

K. Power Consumption Overview - Dieter

The drone port is expected to have various systems continuously powered for remote access, including the main computer and networking hardware. Understanding the device load is pertinent for solar and battery storage calculations. Device loads also vary with idle and active processes. Further testing of devices will be required to accurately represent idle operation.

Depending on networking configuration, the drone port will have varying power demands. Starlink internet provides a more robust network connection in rural areas with little cellular reception. The cons towards this implementation are a higher power draw at load and idle, with additional hardware needed for DC conversion of the Starlink hardware.

Device	Input Voltage	Idle Current (Amps)	Idle Power (Watts)	Max Current (Amps)	Max Power (Watts)
Router	5VDC		TBD		1
4G Modem	5VDC		TBD		2
Starlink	120VAC	0.5	20	2	75
Beelink Computer	19VDC	~0.5	~10	3	60
Raspberry Pico	5VDC / 3.3VDC	0.05	0.25	0.1	0.5
DC Motors	12VDC	TBD	TBD	3	36
GPS Module	5V	TBD	TBD	0.15	0.75
Radio Module	5V	TBD	TBD	0.8	4
Misc. Sensor Devices	5V	TBD	TBD	0.4	2
		TBD	TBD		

Figure 30: Power Consumption by device.

Operation	Current (Amps)	Power (Watts)
Full on Starlink	10.45	183.25
Full on 4G LTE	10.45	118.25
Active Starlink (Estimated)	7	80
Active 4G LTE (Estimated)	6	70
Idle Starlink (Estimated)	3.5	40
Idle 4G LTE (Estimated)	3.5	30
Peak Sun 50W Panel	2.69	50
Peak Sun 30W Panel	1.53	30
Partial Sun 50W Panel (30%)	0.806	15
Partial Sun 30W Panel (30%)	0.513	10
Peak Sun 3x50W Panels	8.07	150
Peak Sun 3x30W Panels	4.59	90
Partial Sun 3x50W Panels (30%)	2.418	45
Partial Sun 3x30W Panels (30%)	1.539	30

Figure 31: Current and power statistics for operating modes.

Operation	Daily Activity (Hours)	Current (Amp Hours)	Power (Watt Hours)
Full on Starlink	4	41.8	733.6
Full on 4G LTE	4	41.8	473.6
Active Starlink (Estimated)	4	28	320
Active 4G LTE (Estimated)	4	24	280
Idle Starlink (Estimated)	20	70	800
Idle 4G LTE (Estimated)	20	70	600
Peak Sun 3x50W Panels	4	32.28	600
Peak Sun 3x30W Panels	4	18.36	360
Partial Sun 3x50W Panels (30%)	4	9.672	180
Partial Sun 3x30W Panels (30%)	4	6.156	120

Figure 32: Daily power generation and expenditure for operating modes

Drone port power consumption can be observed in active and idle operation modes. This allows for the main computer to process Ortho-map data, upload data to the cloud, and charge the drone. A large majority of the day the drone port will be in idle mode, reducing power consumption as much as possible. While further testing is necessary, estimation of a 24-hour cycle shows ~100Ah battery necessary and at least a 150W panel solution for field operation.

L. ADC Configuration - Dieter

Since the ADC is on the Raspberry Pico, initial setup of the ADC can easily be achieved using the ADC section of the Pico Software Development Kit (SDK) [14]. This provides the engineer with a simple and straightforward introduction on taking ADC readings. However, the 12-bit ADC onboard the Pico is not great when unaltered with 8.7 as the effective number of bits [15].

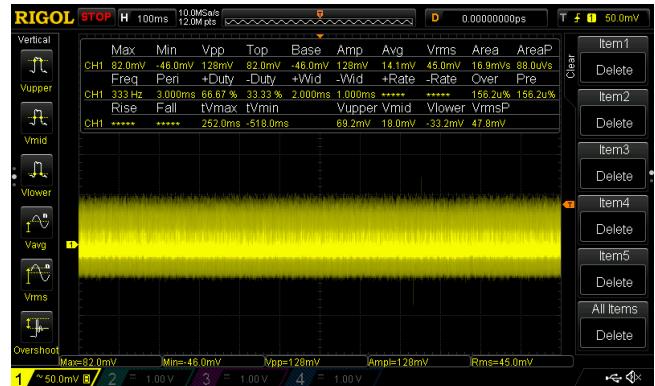


Figure 33: Pico 3.3V Noise Measurement (20MHz BW limit, 10x probe scaling, and ground spring attached).

This is due to the switching power regulator on the Pico which can have as much as 200mV Vpp of noise. As a result of switching noise, the onboard voltage reference for the ADC is setup in poor conditions. There is an RC filter feeding the ADC reference which creates a 30mV offset with a slightly less noisy signal [15]. The datasheet gives suggestions on improvement of the ADC readings. An External reference voltage may be used, the R7 resistor can be removed, or issues can be mitigated in averaging and offset code.

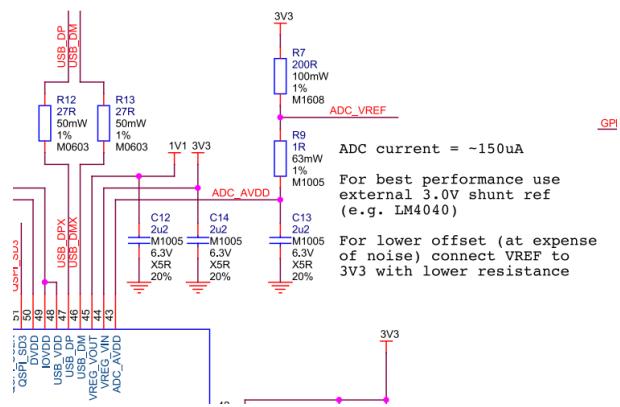


Figure 34: Pico ADC Connection [15].

We chose to disconnect the ADC reference from the Pico power chain entirely by removing the R7 resistor from the microcontroller. Then a reference could easily be made using a TLV431 adjustable voltage reference. Since we are designing this circuit for low power consumption, we instead chose to

implement a LM4040 3V micropower voltage reference [11]. This was fed from the 5V rail using a 1k ohm resistor and filtered using 100nF decoupling capacitors. In addition to feeding the ADC, this 3V reference was buffered and used for the creation of precise voltage references used elsewhere.

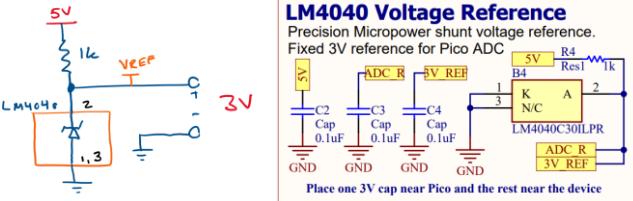


Figure 35: ADC LM4040 3V Reference circuit

M. Voltage References - Dieter

Precise voltage references are desired for comparison circuitry throughout the power measurement system. The most prominent voltage references required are for the current sensing system. A 2.5V reference is used for the bidirectional current sensor, while a 50mV reference is used for the unidirectional current sensors. This allows for the current measurements to remain accurate, especially when the amplifiers operate close to the ground rail.

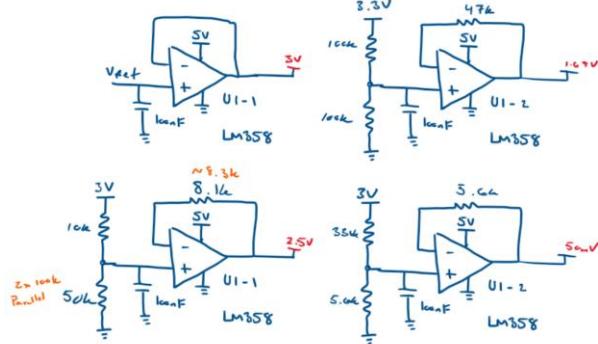


Figure X: Voltage Reference circuits.

References were made using the 5V rail, resistor dividers, and voltage buffers. This allows for the references to be made with high resistances and minimizes current and power usage. The voltage buffers are implemented to create a low output impedance compared to the Thevenin equivalent circuit made by the voltage divider. LM358 op-amps were chosen because they are cost effective (~\$0.10), have relatively low quiescent current (350uA typ, 600uA max), and can operate in the desired voltage range without clipping. We believe that input offset voltage would be the only downside to using this device as 2mV offset is typical and can reach up to 5mV in a worse case. [10]. Feedback resistors were chosen to match the input impedance seen from both the inverting and non-inverting terminals to help mitigate this flaw. Regardless, the circuitry is more than acceptable for generation of these references and their applications, as other limitations to the circuits such as ADC resolution and current amplifier accuracy exist.

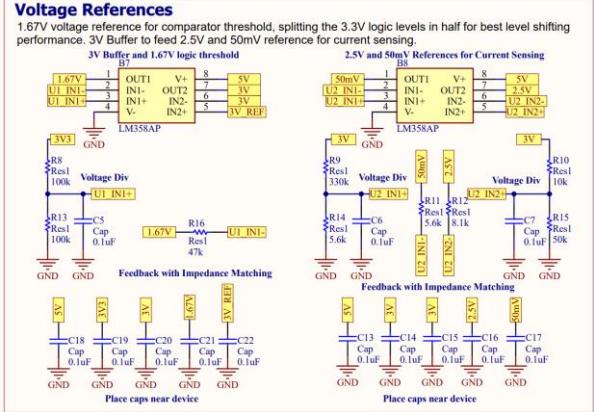


Figure 36: Voltage Reference Schematic.

N. ADC Channel Expansion - Dieter

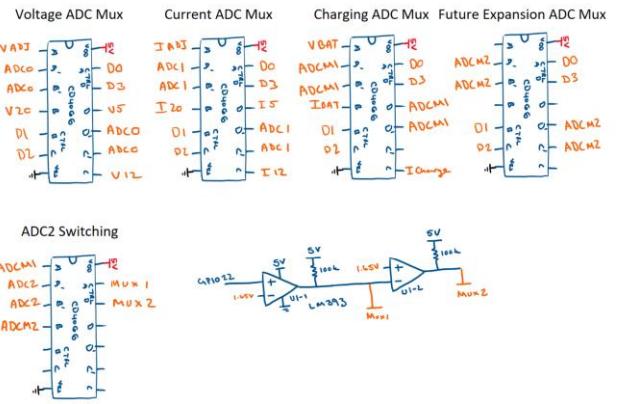


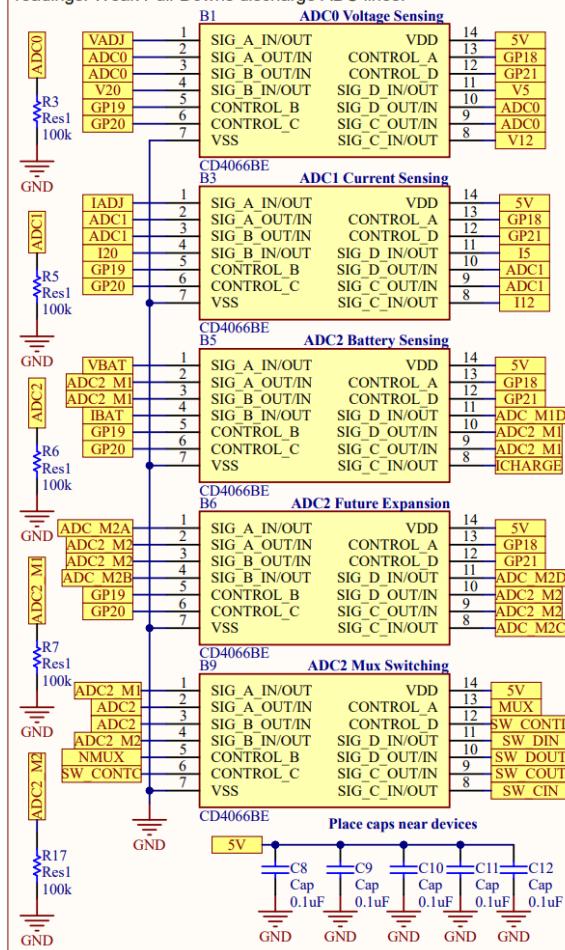
Figure 37: ADC Channel multiplexing circuit.

Aside from conditioning the ADC to take precise measurements, it is necessary to have many ADC channels so that we can take a variety of current and voltage measurements without intervention and crosstalk. While external ADCs could be implemented, ADC's can be relatively expensive and often come in small SMD packaging. While this is an area for improvement and footprint shrinking in the future, we believe that it is more important to have the cost efficiency, debuggability, and general ease of use that is associated with DIP parts. This severely limits the quality of ADC's when searching. As a result, we took a different approach and created a multiplexing circuit to effectively expand the ADC channels on the Pico.

By using the CD4066B Bilateral Switch, we can effectively multiplex analog signals with little interference or distortion [3]. ADC channels 0 and 1 have their own CD4066 IC in which they measure voltage and current of the output rails respectively. ADC2 is shared between a mux that measures the battery and a future expansion mux in the case additional current and voltage sensing is desired for drone charging or motor overcurrent. The ADC2 Multiplexers are switched between by an intermediate switch. The intermediate switch is controlled by a single GPIO so that one CD4066 connected to ADC2 is always on. This signal is created by using a LM393 dual comparator with a middle logic threshold value of 1.67V. The circuit accomplishes

ADC Analog Switching

CD4066 IC's extend the 3 ADC channels to a potential 16 ADC readings. Weak Pull-Downs discharge ADC lines.



two things, it first performs logic level shifting from 3.3V to 5V while also creating two opposite signals from one GPIO pin.

Each mux is controlled by four data lines. A single data line is set high and all three ADCs are queried before setting the data line low again. This process is limited by the Setup and hold time of the ADC, which is approximately 2us from the 500ksps Pico ADC [15]. In python, 20us seems to be a stable setup and hold time for each ADC measurement and allows for every expanded channel to be read in less than 100us. For the use case of this project this rate of readings is more than sufficient.

O. Reverse Polarity Protection - Dieter

Protecting against reverse polarity conditions helps prevent hardware damage due to user error. This is a safety feature that should be included in our circuit in an event where a battery or solar charger is connected in reverse. This can be accomplished by implementing a power NMOS or PMOS in series with the

Figure 38: ADC multiplexing schematic power signal. For

this design I chose to implement PMOS to avoid interrupting the ground reference as a good practice. This system can also be achieved using a NMOS transistor in a mirrored circuit if interrupting the common ground is not a concern. Assuming the correct connection, current will initially flow through the body diode until the output voltage rises above the PMOS voltage

threshold. Once this occurs, the MOSFET will start conducting and provide a low impedance path for current to flow that corresponds to the $R_{ds\text{ on}}$ of the MOSFET. This allows for the voltage at V_{in} and V_{out} to be essentially the same value. In the event that the connection is incorrect, the body diode will not conduct and the V_{gs} of the MOSFET will remain at zero, effectively maintaining a disconnected state.

Low Voltage Reverse Polarity Protection

Voltage limited by V_{gs} of PMOS, current limited by physical junctions.

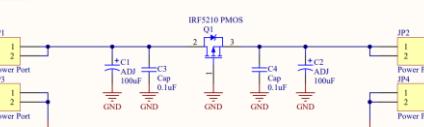


Figure 39: Simple reverse polarity protection circuit. V_{in} is limited to the V_{gs} of the PMOS.

In our circuit we use an IRF5210 which has an $R_{ds\text{ on}}$ of ~60m Ohms, making the drop over the internal resistance negligible when active [8]. Heat dissipation will occur over this internal resistance, for applications over 1W it is crucial to apply a heatsink to the TO-220 package device. For the IRF5210, we advise using a heatsink when having currents more than 3A, as the power dissipation on the device is equivalent to $P = I^2R$. Decoupling capacitors are added to minimize transient spikes in power usage.

Reverse Polarity Protection

Voltage limited by V_{gs} of PMOS, current limited by physical junctions.

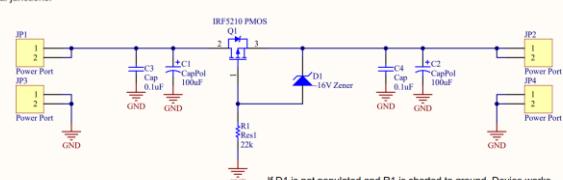


Figure 40: Reverse Polarity Protection circuit.

There is one drawback to the first circuit, as the input voltage can never be more than the V_{gs} allowed by the MOSFET. If the V_{gs} is exceeded, the device is likely to fail and potentially damage downstream electronics. However, with some simple additions to our circuit we can increase the maximum input and output voltage. A Zener diode can be added between the gate and source terminals to regulate the V_{gs} within a certain voltage level. A resistor is also added to provide a connection to ground and provide a difference between the gate and ground, as the gate voltage will rise above ground when the source voltage surpasses the Zener diodes voltage. For the IRF5210, the maximum V_{gs} is -20V [8]. The IRF5210 is also the most active when the V_{gs} exceeds -10V. As a result, we chose to implement a 16V Zener diode to keep the maximum V_{gs} of the circuit at -16V. This also allows for substitute Power PMOS to be utilized that may have a similar maximum V_{gs} with different thresholds to remain saturated. Choosing the resistance of R_1 is a balance of quiescent current consumption and reaction time. A large resistance could be placed here for less quiescent current at the

expense of slower reactions to the input switching polarity, this can be an issue if large capacitance is seen on the output of the circuit.

P. Output Power Switch - Dieter

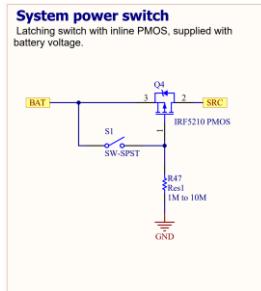


Figure 41: Simple PMOS power switch circuit. The PMOS is conductive when the S1 switch is open.

Similar to how the reverse polarity circuit operates, an output power switch circuit was incorporated to shutoff power from the Battery and charger circuit to the output regulators. This effectively turns off anything attached to the system but does not interfere with the charging of the battery. A switch that is active when open is connected between the Battery source and gate of a PMOS device. This PMOS device also has a large resistance between its gate and ground as a weak pull-down. When the switch is closed, V_{gs} become zero and the MOSFET no longer conducts. When the Switch is open, V_{gs} can increase to the difference between the battery voltage and ground, allowing large amounts of current to conduct.

Q. Overvoltage Protection - Dieter

While many of our DC/DC converters are equipped with over voltage and over current protection features, it is a good practice to incorporate some protection circuitry for the converters that do not include protections and provide a low resistance path in the event of an overvoltage condition. This can be done in a variety of ways such as overvoltage crowbars or voltage clamping. Overvoltage crowbars effectively trigger at a threshold well above the operating voltage of a rail. This is because crowbars create a shunt to ground when triggered to pop a fuse somewhere in the circuit. This isn't an ideal situation for our power chain as drawing high amounts of current and replacing fuses is a brute force implementation of protection, and other electronics can be damaged in the time it takes for the fuse to break. Therefore, it makes more sense to provide voltage clamping to regulate the voltage if it crosses a specific threshold.

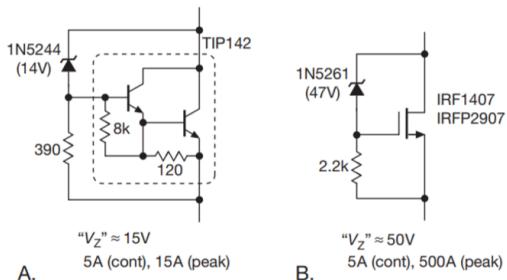


Figure 9.103. Active power zener.

Figure 42: Active Power Zener for Voltage Clamping [6].

Voltage clamping can be easily accomplished using a Zener diode for low current systems. However, as many of the DC/DC converters in the circuit can output current in the 5-10A range, a normal Zener diode would almost immediately fail under such high current. For that reason, we decided to implement active power Zener clamps. In this circuit, a Zener diode will start conducting at its internal voltage threshold, but a pass transistor is used to sink the higher currents of the system. This can be done with a power BJT or MOSFET and certain transistors make more sense depending on the desired voltage threshold and the current being shunted. A BJT, which is often configured as a Darlington pair as seen with the TIP142 in the above figure, will become active when the base of the transistor is two V_{be} 's above ground or approximately 1—1.4V. This is desirable for low thresholds. The main drawback of the power BJT's in this circuit is that they cannot sink much current. In contrast to the above figure, TIP142 has a maximum continuous current specification of 10A [17]. When a MOSFET is used for this sort of system, activation thresholds are much higher and MOSFETs can often sink more current.

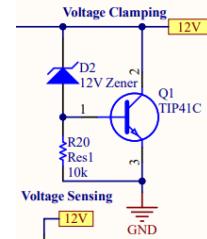


Figure 43: Example Voltage Clamping Circuit on 12V rail.

We chose to implement active power zeners using BJTs for their simplicity in understanding the cutoff thresholds and their ability to shunt the current that the system will provide. For single BJTs such as the TIP41, the activation level of the circuit will be around 12.6V. This occurs because the voltage at the base will rise when the diode starts conducting above 12V. When the base voltage increases towards 0.6V, the BJT will start conducting and limiting the rail voltage. While this system does waste current over the BJT, it is a better practice to waste power than to destroy attached electronics.

R. Voltage and Current Measurement - Dieter

Monitoring various current and voltages is pertinent for precise power monitoring and smart management of energy in a battery-operated system. While voltage measurements are as simple as measuring the potential difference between two points, measuring current is a bit more complicated. Current can either be measured by sensing the magnetic field induced by a given current, or by measuring a potential difference across a known resistance. The latter is often better for precise measurements of small currents. Current measurement schemes often incorporate a very small resistance to minimize voltage drop over the known resistance. A difference amplifier is then used with high gain to create a measurable voltage as a ratio to current.

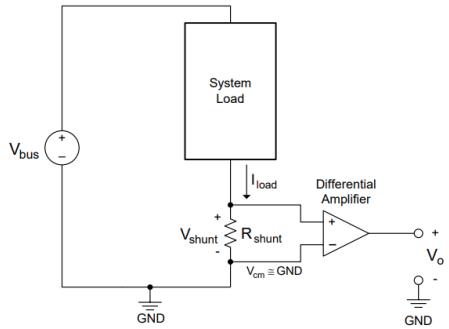
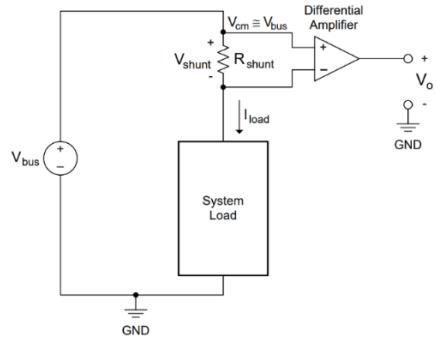


Figure 44: High and Low side current sensing schemes [4].

Current measurement can take place at the high or low side of a system load. Current sensing on the low side of a circuit is easier and cheaper to implement because the common mode voltage is effectively zero. However, low side current sensing interrupts the ground reference for the load and cannot detect if the load has been accidentally shorted to ground, bypassing the shunt resistor. High side current sensing can detect load shorts to ground as it monitors directly from the source. Measuring current from the high side does have drawbacks in the form of high common mode voltage. This often requires more expensive differential amplifiers when compared to low side sensing.

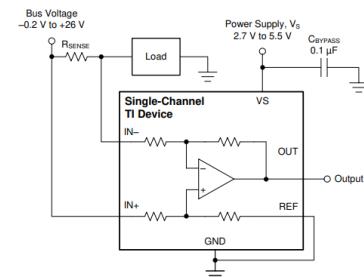


Figure 44. Unidirectional Application

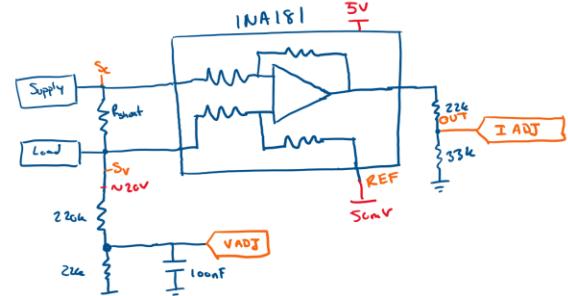


Figure 45: Unidirectional High Side Current Sensing with INA181 and implemented circuit [9].

We chose to implement an INA181 current sense amplifier IC for our current measurements. In most rail measurement applications, a 5m ohm shunt resistor was used in combination with a 200-gain version of the INA181 IC [9]. This allows for a 1:1 ratio between current measured across to precisely measure 0-5A. While the INA181 is considered rail-to-rail, amplifiers are non-ideal and can only get so close to their supply rails. In the case of unidirectional measurement, a 50mV offset is incorporated to measure zero current conditions. Current and voltage measurements are divided to remain on a 0-3V scale before being sent to the ADC system.

In the case of monitoring the battery and charging system, we chose to implement a similar circuit using the INA181 configured in a bidirectional mode. An offset of 2.5V is applied to the INA181 to center the output of the current amplifier in its

Battery and Charger Connections

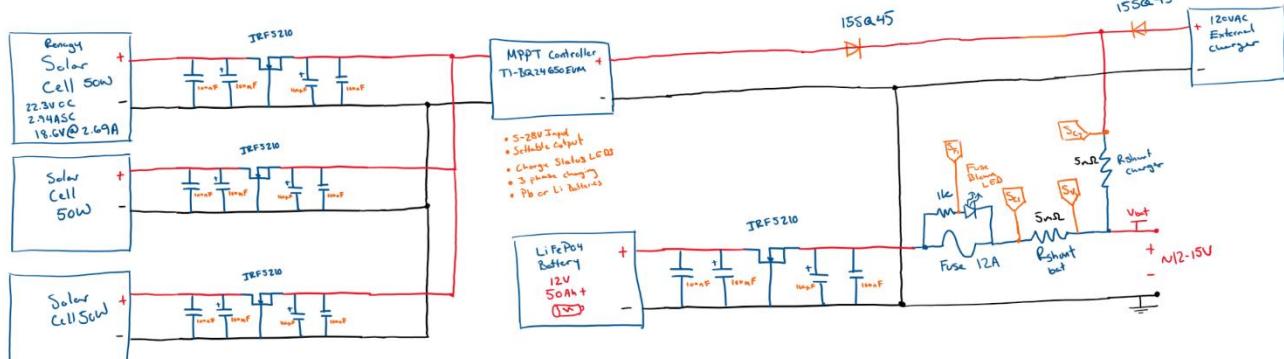


Figure 46: Battery Charging Circuit.

operating region. A 50V/V gain version of the INA181 is utilized so that a ratio 0.25V:1A is achieved [9]. This allows for ~10A to be monitored in either direction. Similarly, the voltage and current measurements are scaled before being sent to the ADC system.

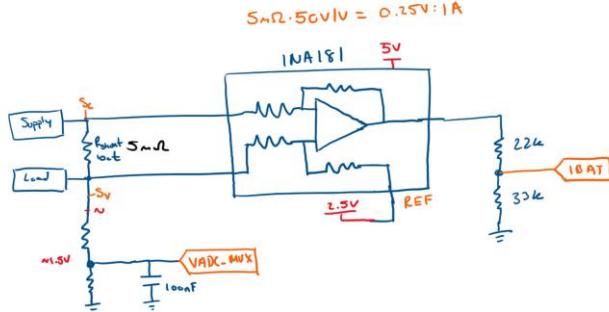


Figure 47: Bidirectional High Side Current Sensing with INA181 for battery charging and discharging.

S. Battery Charging Circuit - Dieter

Charging the battery of the drone station requires careful circuitry to manage the charging. Solar panels are connected using the reverse polarity circuit to a maximum power point tracking (MPPT) controller. For this controller, we found an evaluation module (EVM) from TI that accepts a 5-28V input, resistor-programmable output, LED charge indication, and is compatible with the LiFePO₄ battery chemistry.

Chargers are connected to the same charging node via high current Schottky diodes, allowing for both external and MPPT controller system charging without reverse current damaging the chargers. All charging energy is either directed towards the battery or the load circuit depending on current demands. One shunt resistor is used to monitor battery charging and discharging while the other monitors peak charging output. A glass fuse is placed in the current path along with an LED indicator for a blown fuse. The battery will also include a reverse polarity protection circuit to prevent damage to downstream circuitry.

T. Battery Monitoring – Dieter

To help achieve the goal of smart power management for the base station, we interpreted the voltage and current measurements to create a battery monitoring system using State of Charge (SoC) tracking. After finding an initial charge estimation upon startup, the battery monitoring system can efficiently monitor the current in and out of the battery over time, also known as coulomb counting. We implemented a system of SoC monitoring using an Analog Devices SoC tracking algorithm that tracks Battery health, state of charge, and depth of discharge [13].

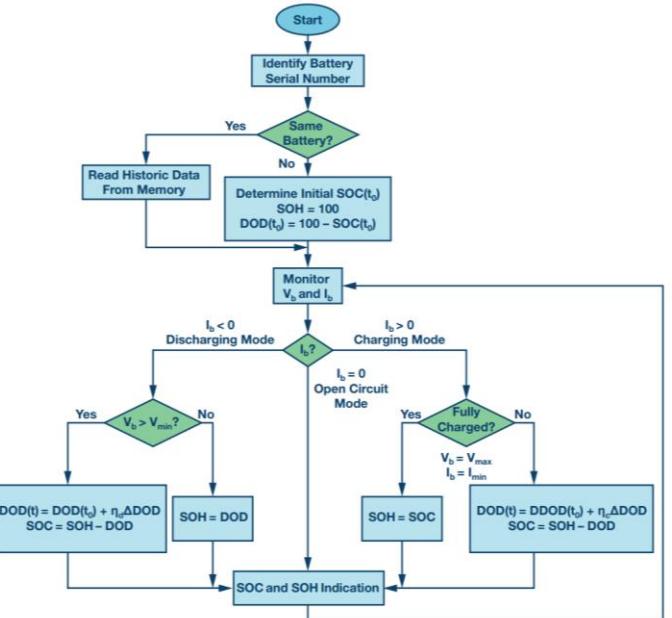


Figure 48: Coulomb Counting Algorithm [13].

U. BQ24650EVM MPPT Charge Controller – Dieter

The BQ24650EVM is a highly integrated switch-mode battery charge controller made by Texas Instruments that provides input voltage regulation for solar panel support [2]. Features include PWM control, High accuracy voltage and current regulation, charge preconditioning, charge termination, and charge status monitoring [2]. Effectively, this device allows for constant-current, constant voltage (CCCV) charging for 2.1 to 26V batteries while also acting as a maximum power point tracking (MPPT) controller for efficient charging via solar.

Table 3. Recommended Operating Conditions						
Symbol	Description	Min	Typ	Max	Unit	Notes
Supply voltage, V _{IN}	Input voltage	5	20	28	V	
Battery voltage, V _{OUT}	Voltage applied at VOUT terminal of J2	2.1	12.6	26	V	
Supply current	Maximum input current	0		8	A	
Charge current, I _{CHG}	Battery charge current	0	2	8	A	For charge current above 2 A, replace R6 and L1 with high-current rating components
Operating junction temperature range, T _J		0		125	°C	

Figure 49: BQ24650 Operating Conditions [2].

Figure X:

The BQ24650EVM is initially set to deliver 2A of charging current at 12.6V. For our application we desired a 5A current limit with approximately a 14.5V voltage. This will allow us to safely charge the battery without stressing the evaluation board or the battery. In future implementations of this system, designers may desire to have the maximum current output possible for this board. To change the output voltage and current limit, we referenced the equations of the BQ24650EVM datasheet and found equations for editing the desired parameters. As a result, we chose to edit the R13 and R6 values to 590k and 8m ohms respectively.

$$V_{\text{OUT}} = 2.1 \text{ V} \times \left(1 + \frac{R_{13}}{R_{15}}\right) \quad V_{\text{MPPSET}} = 1.2 \text{ V} \times \left(1 + \frac{R_{17}}{R_{19}}\right)$$

$$I_{\text{CHARGE}} = \frac{40 \text{ mV}}{R_6}$$

Figure 50: BQ24650EVM Equations [2].

V. TPS1297AEVM DC/DC Buck Converter - Dieter

The TPS1297A is a synchronous buck converter made by Texas Instruments that can deliver up to 10A output [19]. The EVM is capable of input voltages between 5.5 and 24V and will be supplied by the battery of our system. From the factory the board is set to deliver 5VDC output and has a 30mVpp ripple voltage when fully loaded. This board was chosen for its 90%+ efficiency, great transient response, relatively low noise, and large current delivery. The EVM performs with greater efficiency at larger current loads. This allows for future expansion of devices and systems that require a 5V source. No edits were necessary to have the desired functionality of the EVM.

W. I2C TPS55288EVM DC/DC Converter and Driver - Dieter

The TPS55288 IC is a programmable Buck Boost converter with an I2C interface made by Texas Instruments. TI sells an evaluation module (EVM) of the device for research, test, and design purposes such as ours. The IC is designed for USB power delivery specifications (USB-PD). It has a wide input voltage range between 2.7V and 36V paired with an output range of 0.8V to 22V [20]. Voltage steps can be taken in 20mV increments over the entire output range, or smaller steps for limited output ranges, allowing higher resolution voltages. The device has a programmable current limit up to 6.35A with 50mA increments within a 5% tolerance [20]. The datasheet boasts 97% efficiency when under conditions of a 12V input and 20V/3A output. These DC/DC converters were chosen for their input and output characteristics, alongside their high efficiency, EMI mitigation system, rich protection features, and overall small size when implemented [20].

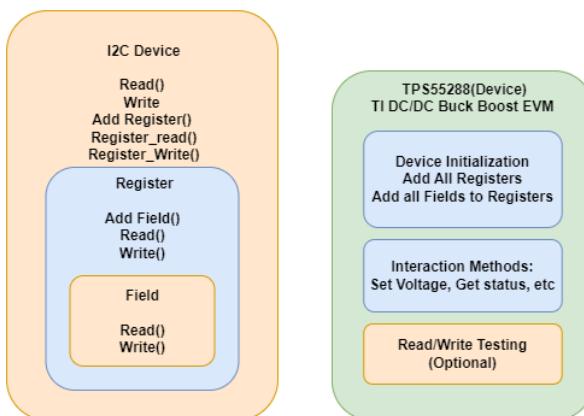


Figure 51: TPS55288 Device Driver Register Mapping code structure.

While the device has impressive specifications, the incorporation of an I2C interface requires that master microcontroller exist to command parameters of the DC/DC

converters on startup. The raspberry Pico will act as our master device and initialize each of the TPS55288EVM's upon startup. These EVM's are used to create three rails with unique purposes. One rail designated as the VADJ was designed specifically for drone charging. This can be used on its own and provides constant current and constant voltage regulation for attached battery charging. The second rail this EVM supplies is designated as the 20V rail and was designed specifically to power an attached computer up to 20VDC/5A. With the Beelink computer, we set this rail to 19V to better match the power supply that came with the Beelink. The last EVM is designated as a 12V rail designed for attached motors and future expansion such as the merging of the sentinel project within the base station.

X. TCA9548 I2C Switch - Dieter

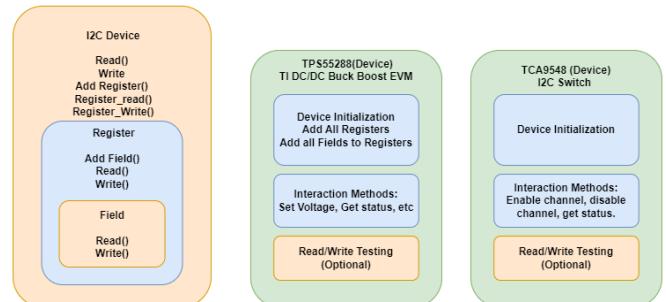


Figure 52: I2C device and specific Driver Code structures

Issues of I2C addressing arise when using multiples of the same device on a single I2C bus. In our case, we have three devices that share the same address. While the addresses can be programmed internally on the TPS55288, device memory is reset upon a power cycle and would need to be set differently each time the system is powered on. The TPS55288 can also be programmed to a single alternate address externally, however this requires editing the EVM hardware and does not provide enough alternate addresses for all three devices being implemented. Therefore, we chose to add a simple I2C switch such as the TCA9548 to the design to swap between channels of communication without needing to edit the end device's I2C address [16]. The I2C switch simply opens or closes channels of I2C communication, effectively acting as a digital switch system. Since speed of communication is not a constraint the TCA9548 is sufficient for this system. The TCA9548 also provides 8 channels for switching and allows for communication over multiple channels simultaneously [16]. The device is also tolerant to various logic levels and can be easily found in breakout boards.

Y. DRV8871 H-Bridge Driver - Dieter



Figure 53: DRV8871 H-Bridge Breakout Board.

The DRV8871 is a CMOS H-Bridge Driver IC manufactured by Texas Instruments. These boards are commonly available in Breakouts by Adafruit and other manufacturers. The DRV8871 is a versatile H-Bridge driver that can operate between 6.5V and 45V and supply up to 3.6A [5]. Control of the DRV8871 can occur with both 3.3V and 5V logic with static and PWM control schemes [5]. Current limiting is determined by an external resistor, enabling overcurrent protection on the device. This specification proves useful for both driving motors on the station but also for the drone charging circuit discussed later.

Table 1. H-Bridge Control

IN1	IN2	OUT1	OUT2	DESCRIPTION
0	0	High-Z	High-Z	Coast; H-bridge disabled to High-Z (sleep entered after 1 ms)
0	1	L	H	Reverse (Current OUT2 → OUT1)
1	0	H	L	Forward (Current OUT1 → OUT2)
1	1	L	L	Brake: low-side slow decay

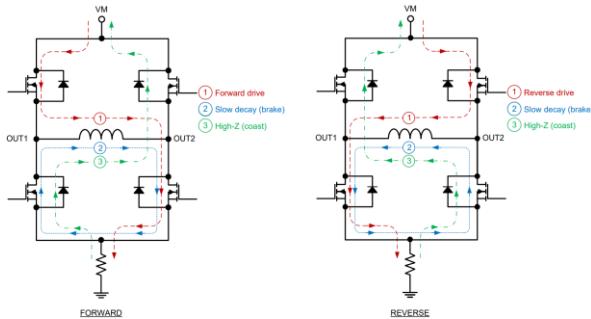


Figure 4. H-Bridge Current Paths

Figure 54: DRV8871 control logic and current paths [5].

We developed a micro python driver for the DRV8871 that allows for PWM control of the H-bridge using two GPIO pins from the raspberry Pico. Given correct wiring, the microprocessor can control the flow of current and the duty cycle to which it is pulsed. As a result, a developer can easily command the H-bridge into a forward, reverse, coast, or brake state. The developer can also determine the duty cycle for which the system should be active and if the H-bridge reverts to a braking or coasting state when off.

Z. Drone Charging - Dieter



Figure 55: Drone station landing pad.

To charge the drone that lands on the station, we chose to implement a conductive charging method for efficiency and to minimize weight of charging circuitry on the drone. Agribugs generously provided a prototype landing pad for drone point charging. This landing incorporates four metal pads in each quadrant of a circle. To develop charging using this landing pad, a few assumptions were made to streamline development. Our charging solution assumes that the drone can safely and correctly land on the pad without errors in orientation. The second assumption is that the batteries that are attached to the drone can balance their own cells. While a small and lightweight safe charging board can be developed to assist with this, onboard battery management for the drone was not in the scope of development.

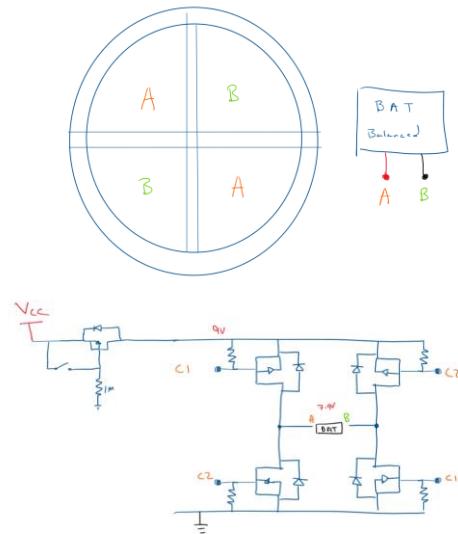


Figure 56: Drone station landing quadrants and H-bridge.

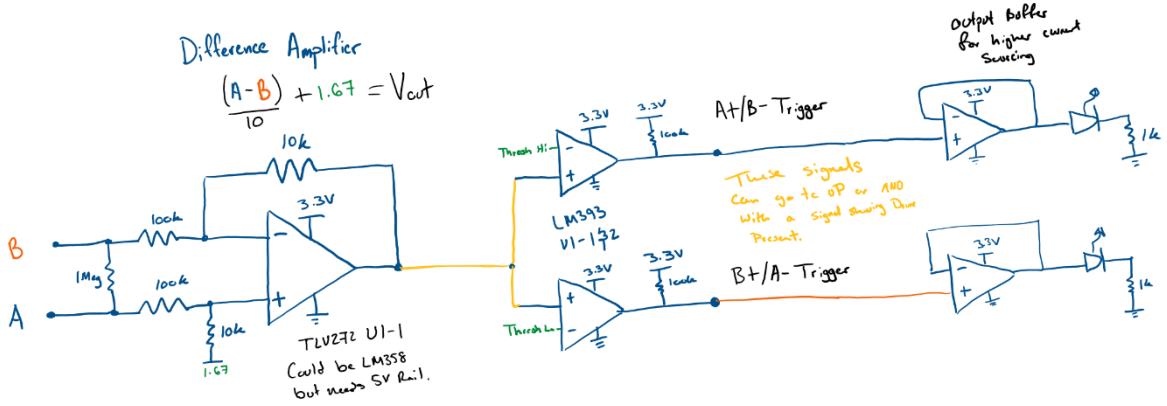


Figure 57: Battery Polarity Detection Circuit

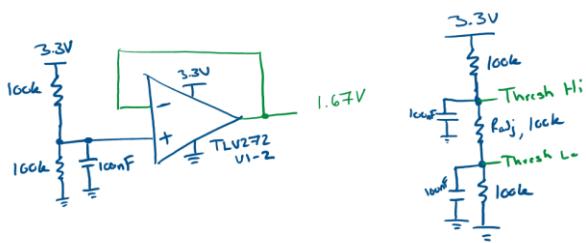


Figure 58: Threshold setting circuitry for the polarity detection circuit.

By dividing the charging pad into quadrants and having charging points of the drone battery tied to alternating quadrants, there can only be two possible configurations where the battery is connected. As a result, only swapping between two different current paths instead of four is required for connection of the battery circuit. This Swapping of current paths can be done with an H-bridge device like how many motors can be driven in both directions. The biggest difference being that for charging you would provide a static setting instead of PWM for the motor control to allow the maximum voltage and unhindered current flow. This is efficient for charging the attached battery as the only losses seen are due to the internal resistance of two H-bridge MOSFET's.

The drain to source resistances when on (R_{ds_on}) are usually small, and in the case of the DRV8871, approximately 565 milliohms in total. Power dissipation on a device is equivalent to $P = I^2R$, meaning that for 1A being supplied to a charging battery, there is approximately 0.565W of waste energy when delivering the current using the DRV8871. The overall efficiency of the system depends on the speed of charging and the voltage needed to charge the attached battery. Given a 12V battery and a 14V charging voltage with a 1A current limit, the system would be 96% efficient. More losses occur at higher currents and lower voltages. Also, the DRV8871 was not meant to be used in this situation. A different H-bridge driver could be chosen, or a custom H-bridge circuit could be developed for lower R_{ds_on} and higher efficiency. We chose to use the DRV8871 because of its small form factor, variety of operating voltages and currents important for universal usage, and the easy-to-use control scheme at varying logic levels.

To determine the polarity of the attached battery when contact is made, we chose to implement a bespoke polarity detection circuit. When battery voltage is sensed at terminals A and B, the difference of the two points is observed and compared. A difference amplifier with voltage offset and a gain of 1/10 works to scale down battery voltages to a value within the 0-3.3V range seen by the difference amplifier. Since the scaled down difference is offset to the center of the op-amp's operation region, comparators can be used to determine directionality of the circuit using equal and opposite thresholds. For instance, if a 9V battery is attached with A as positive and B as negative the output of the difference amplifier would be 1.67V + 0.9V or approximately 2.6V. With thresholds set at 2.2V and 1.1V like in the above figure, the top comparator would go high and show the A+/B- trigger while the bottom comparator would be low. The thresholds can be adjusted by swapping a resistor in the attached voltage divider.

We chose to implement this with TLV272's because of their rail-to-rail output operation but the circuit could be adapted to LM358's for cost efficiency if given enough voltage headroom [18]. We desired to drive the entire circuit on 3.3V and as a result needed rail-to-rail operation and TLV272 are considered inexpensive in this category. TLV272s are approximately \$1 for a dual package op-amp. LM393s perform well in this system for strictly driving the digital logic at responsive speeds. Additional buffers are required for supplying current to diagnostic LEDs. These op-amps could be eliminated entirely if the raspberry Pico managed the LED driving or LEDs were omitted entirely.

Simulation of this circuit yielded promising results for detecting and delivering power to the attached battery. Attempts were made to have the entire system control done using analog design, however detection of the battery leaving the circuit proved difficult to implement without a timer system or added current measurement. Ultimately, the circuit H-bridge control was done using the raspberry Pico and charge shutoff was implemented using a timer system for simplicity.

IV. RESULTS

A. PCB Assembly - Dieter

Assembly and debugging of the printed circuit board occurred with minimal issues. There are minor bugs with the board, with one battery output header being routed incorrectly and Thermal relief being too large making ground solder points

difficult to complete. Overall, the board worked well for the intended purposes and the minor issues found were easily solved or were designed with redundancy. EVMs were wired and glued down to the board to avoid movement. Safety and debug features of this board performed well. The biggest issue with this board is the large size. While this is a valid critique, DIP packaged ICs, through hole components, debug headers, and EVMs used can easily be swapped out for more compact versions of the same parts. This will most likely reduce the size of the PCB by more than half when revising the board.

B. ADC Configuration - Dieter

Isolation of the ADC reference and supply the ADC with an external 3V reference was effective at reducing reference noise. While the Rigol scope used for measurement is not the best, there is consistent measurement techniques of 20MHz bandwidth limiting, smallest ground loops possible, and 10x Probe scaling. There is an observable decrease in Rms and Peak-to-Peak noise levels between previous and new ADC references. Approximately a 100mV decrease in Peak-to-Peak noise and 35mV decrease in Rms noise. This correlates to more precise measurements when using the ADC system and higher ENOB.

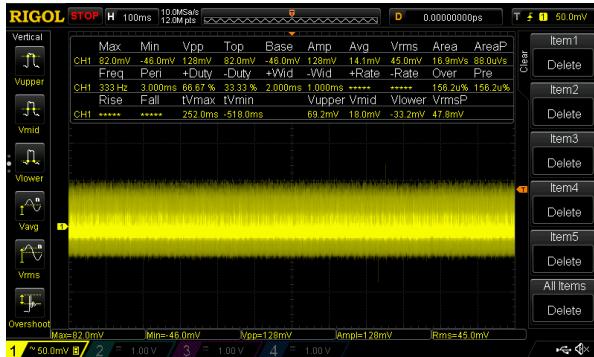


Figure 59: Pico 3V3 Noise Measurement (20MHz BW, 10x Probe and Scaling, Ground Spring attached).

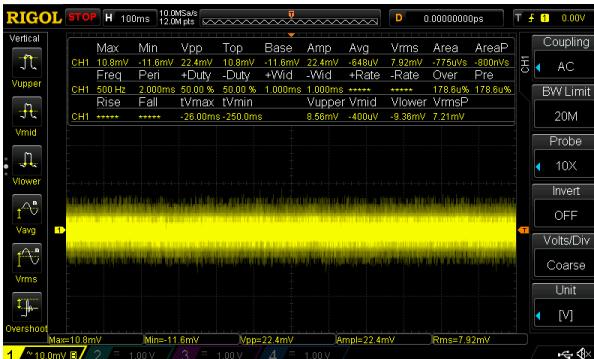


Figure 60: LM4040 3V Noise Measurement (20MHz BW, 10x Probe and Scaling, Ground Spring attached).

C. Voltage References - Dieter

Every voltage reference operated within a typical offset voltage error of 2mV. There were some errors in layout for the voltage references, as R15 should have been two 100k resistors in parallel. Also, the 50mV reference would raise in voltage over time, presumably due to parasitic capacitance of the circuit board. To mitigate this, the output capacitor C17 was replaced with a 100-ohm resistor to force the op-amp to consistently

source small amounts of current and regulate the output. Overall, the voltage references performed well considering the cost of parts and use case.

D. ADC Channel Expansion - Dieter



Figure 61: ADC Channel Expansion Readings.

ADC channel expansion was implemented seamlessly from prototyping. Measurement data had some inaccuracies due to cold solder joints to the ground pour on the PCB. This issue was found and remedied quickly, allowing for accurate measurements of both voltage and current. In the above figure, we observe the three ADC channels and the 3.3V rail. The multiplexing system effectively switches the connection for each ADC channel simultaneously then takes a reading for each. This occurs four times for a total of 12 measurements in a single polling of the ADC system. Voltage measurements of ADC0 correspond to Channel 1 of the scope. Current measurements of ADC1 correspond to channel 2 of the scope. Lastly, the battery and charging measurements of ADC2 are on channel 3. Channel 2 experiences larger scaling because the currents seen by the system were practically nothing. However, offsets of the current sense amplifiers can be observed when the channel is opened using the multiplexing system.

E. Reverse Polarity Protection - Dieter



Figure 62: Reverse Polarity protection during polarity swapping event without a load.

The reverse polarity protection board is effective in its main purpose of protection. Given a reverse polarity event with no load, the circuit discharges the onboard capacitors and remains within a diode drop of the ground reference. This shows that the

circuit is working as intended, however sudden capacitor discharge may prove troublesome for sensitive electronics. As a safety feature, this circuit seems to work nicely, If there were repeated reverse polarity events we would recommend a smaller resistance or smaller capacitor bank.

F. Overvoltage Protection - Dieter

Overvoltage clamping was tested observing output voltage and base-emitter voltage on the shunt transistor. Since the onboard systems have many protection systems in place, this feature was a redundancy for safety in development. We chose to not populate the board with the overvoltage clamping system for efficiency. Since many of the DC/DC converters are programmable, incorporating voltage clamping on rails powered by them would hinder their ability to be refactored in future development. The 5V rail incorporated this system but its inclusion on other rails was ignored and not populated. Given a 4.7V Zener, conduction starts occurring but does not fully activate the clamp until the base-emitter voltage reaches approximately 1.4V. This clamps the voltage to 5.6V without issue.



Figure 63: Overvoltage protection clamping on 5V rail. (TIP142, 4.7V Zener, 22k resistor)

G. I2C TPS55288 DC/DC Converter Driver – Dieter

Control of the programmable DC/DC converters using the I2C device system is streamlined and developer friendly. Outputs of the DC/DC converters are accurate to their programming and quite precise. To test the extent of the EVM and drivers capabilities, A test script was implemented to iterate from 20V to 3.3V in various and sporadic steps. This can be seen in the following figure as the output of the DC/DC converter is set to 20V, 19V, 12V, 9V, 5V, and 3.3V over a period of a second for each step. Every setting of the EVM can be queried

and relays accurate information for the DC/DC converter control system.



Figure 64: TPS55288EVM Test script iterating through various output voltages.

H. Voltage and Current Measurements - Dieter

The voltage and current measurement system works well for the desired use case. Voltage and current readings are well within a 5% accuracy tolerance. There are potential design flaws that could be revised going forward to improve accuracy and precision, However, the implemented measurement system is functional and safe. Large thermal relief on the ground plane makes solder connections poor for capacitors on the ADC measurement system. This can be remedied on a case-by-case basis, or the thermal relief can be smaller in the future. Maintaining the system in their linear region became a balancing act. As a result, edge case measurements show the most inaccuracies. Improvements to the system could be made by removing the weak pull-down resistors on the multiplexing system and instead implementing a transistor that can be activated and provide a short to ground to quickly discharge the ADC lines between multiplexed readings. This would be like how ADCs operate internally. Alternatively, a discrete ADC may be chosen to reduce the footprint of the ADC system onboard and simplify the systems signal development at the expense of higher cost.

I. Drone Charging - Dieter

The drone charging system works as designed and simulated. A DRV8871 device is programmed for GPIO pins 12 and 13 on the raspberry Pico with the polarity sensing circuit giving the A+ or B+ triggers to GPIO pins 10 and 11. This allows for our dummy drone to make contact on the landing pad and signal to the Pico it's orientation. This orientation data is then used in controlling an H-bridge, which spins a motor for demonstration purposes.

J. Agisoft Orthomap Processing - Ryan

The Agisoft script for processing photos works as expected. An actual Metashape license needs to be used, and users are required to have Python 3.8.10 installed. For demonstration purposes, (without a license), the run.bat file is launched from the GUI, checks for an existing virtual environment, and creates one if it is not available. It proceeds to install the Metashape package from a local .whl file, and proceeds with the script.

```
C:\WINDOWS\system32\cmd.exe
Virtual environment 'venv' already exists.
Metashape 2.0.3 is already installed.
License not found
Found 1 GPUs in 0.089 sec (CUDA: 0.001 sec, OpenCL: 0.087 sec)
No GPU found!
AddPhotos
*****  
Matching Photos
*****  
MatchPhotos: accuracy = High, preselection = generic, reference, keypoint limit = 400000, keypoint limit per mpx = 1000,
tchng = 0
Match Photos Progress: 0.00%
Found 1 GPUs in 0.001 sec (CUDA: 0 sec, OpenCL: 0.001 sec)
Using device: Intel(R) UHD Graphics 620, 24 compute units, 6487 MB global memory, OpenCL 2.1
    driver version: 24.20.100.6346, platform version: OpenCL 2.1
        max work group size: 256
        max work item sizes [256, 256, 256]
        max mem alloc size: 3243 MB
Building SPIR-V kernels for Intel(R) UHD Graphics 620...
Kernels compilation done in 3.833 seconds
Building SPIR-V kernels for Intel(R) UHD Graphics 620...
Kernels compilation done in 1.599 seconds
Kernel loaded in 0.002 seconds
Match Photos Progress: 3.00%
[GPU] photo 1: 1392 points
Match Photos Progress: 6.00%
[GPU] photo 2: 1362 points
Match Photos Progress: 9.00%
[GPU] photo 3: 1479 points
Match Photos Progress: 12.00%
[GPU] photo 4: 1540 points
Match Photos Progress: 15.00%
[GPU] photo 5: 1493 points
Match Photos Progress: 18.00%
[GPU] photo 6: 1515 points
Match Photos Progress: 21.00%
[GPU] photo 7: 1552 points
Match Photos Progress: 24.00%
[GPU] photo 8: 1563 points
Match Photos Progress: 27.00%
[GPU] photo 9: 1489 points
Match Photos Progress: 30.00%
points detected in 5.76 sec
Found 1 GPUs in 0.001 sec (CUDA: 0 sec, OpenCL: 0.001 sec)
Using device: Intel(R) UHD Graphics 620, 24 compute units, 6487 MB global memory, OpenCL 2.1
    driver version: 24.20.100.6346, platform version: OpenCL 2.1
        max work group size: 256
        max work item sizes [256, 256, 256]
        max mem alloc size: 3243 MB
Loading kernels for Intel(R) UHD Graphics 620...
Kernel loaded in 0.002 seconds
Match Photos Progress: 32.40%
```

Figure 65: Agisoft Script that processes an orthomap from an available dataset

The script will add the photos, match and align them, build the depth maps, build a point cloud, and finally build the Orthomosaic. The current code is for demo purposes, the final code should be uncommented in each section – without an actual license, it is not possible to save a Metashape project. Processed datasets are moved to an archive folder, and log files of previous project are added to their corresponding dataset folder.

K. Control Station GUI output - Ryan

The control station GUI worked as expected, with program hangs at a minimum due to efficient thread management. The program was set to build in release mode, and features email and company information embedded into the file details. The GUI is updated only when values are changed to maintain low power usage.



Figure 9: Graphical User Interface as a standalone window

The status section of the GUI shows how the peripherals and status of the different devices. Green indicators mean that the devices are connected and functioning properly, while red indicators indicate the devices are not connected. The status is updated on timers and will have some seconds of delay updating the indicator after disconnection or successful connection of a device. The GUI's text size and placement of components on the screen can be adjusted by changing the window size.

V. DISCUSSION

A. Main Computer Software and Hardware

The main computer software was effective in creating an efficient graphical user interface and interacting with many parallel systems. The software will need to be amended and redesigned to fit additional features of the drone station that were not implemented by our team. GUI and automation processes are portable to any Windows or Linux operating hardware. While the Beelink mini-PC (0.061 TOPS, 0.033 TFLOPS FP32) works well for proof of concept and development, a more powerful and energy efficient computer is recommended for a final design. Within the same power budget of 60W and similar form factor, a much more powerful and expensive Jetson Orin AGX 64GB (275 TOPS, 4.096 TFLOPS FP32) could be used. Alternatively, A Jetson Orin Nano 8GB (40 TOPS, 1.024 TFLOPS FP32) could be used for a similar size and cost of the Beelink while remaining more power efficient (15W) and approximately thirty times more powerful at floating point arithmetic.

B. Power Management board

The power management board was a success as a proof of concept and as a development kit for future engineers. As development of the entire station continues, it would be beneficial to revisit the power management board to address issues addressed in this report while also integrating EVM circuit directly and downsizing the circuit design. This would be most beneficial if other systems were ready to be incorporated into a single PCB, creating a unified station board.

C. Future Additions and Implementations

The agricultural drone station project requires many more systems to work in tandem with the design progress our team has made. Incorporation of dedicated internet modem hardware, GPS/RTK hardware, drone RF flight control hardware, weather sensor hardware, live video transmission, cloud data dumping are all systems desired for the final product.

1) GPS/RTK systems

A GPS/RTK system is desired for this project to achieve centimeter accuracy of drone control and survey data. A relatively inexpensive GPS-RTK module and GNSS surveying antenna can be found from Sparkfun electronics. The surveying antenna was chosen by Agribugs and designed into the center of the landing pad on the drone station. More research is needed to choose the best GPS-RTK for this system but the referenced board in the figure below should prove sufficient.



Figure 66: Sparkfun GPS-RTK (GPS-22660) and Surveying Antenna (GPS-17751)

2) Drone Flight Control

Drone flight control was not deeply explored due to the ongoing search for a candidate open-source drone less than 250 grams. Brief attempts proved troublesome due to a malfunctioning controller with the 3DR IRIS in the possession of our team. As a result, our focus shifted elsewhere. In implementing drone control and communication, an RF telemetry system between the drone and base station is required, such as the RFD 900MHz bundle seen below.



Figure 67: RFD900x-US telemetry Bundle.

While communication between the drone and base station can be established on a physical layer using this telemetry bundle, drone control and automation can be programmed from the main PC using PixHawk flight controller. The PixHawk is a commonly used open-source flight controller that provides a low cost, high-end solution to drone automation.

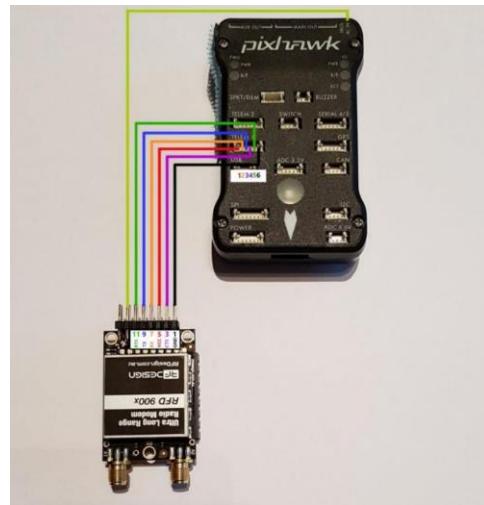


Figure 68: Wiring Diagram of Pixhawk Flight Controller and RFD900x.

3) Internet Services

In the current system, an off-the-shelf travel router was used to create the local area network while a mobile hotspot was used for internet access. Going forward, a 4G-LTE modem or Satellite modem would be ideal for field internet access. A 4G-Modem can be found easily off-the-shelf by companies like Netgear. While a 4G-LTE modem would be relatively power efficient, coverage networks in farmland are unreliable and data rates are often more expensive. Satellite internet could be explored using Starlink or a comparable service, however the Starlink router only accepts 120VAC and as a result will not work on our setup without alteration. There are Starlink community members that can bypass the included Starlink router entirely. A user could create their own 48V power source using a DC/DC converter to supply power over ethernet (PoE). The voltage would be applied to a CAT6 ethernet cable using a PoE injector and would supply the Starlink antenna. The Starlink antenna would act as a modem and can be set up using the Starlink app. The Starlink dish antenna could then be plugged into any router, such as our AC750. This would create a usable satellite internet connection and more affordable data rates with better connectivity. The drawbacks of this system include the added hardware overhead for PoE injection and higher power consumption relative to the 4G-LTE system.

VI. CONCLUSION

Overall, our team was able to effectively kickstart the drone station project by developing a smart power management system and remotely accessible user interface for station data. Wireless communication of drone images, drone charging, and automatic Ortho map computation have been proven in concept. While there are many features desired for the autonomous drone station, the selection of implemented features has been effective and allows future integration and consolidation of the design. Future development can build off the existing design and eventual improvements can be made to favor a smaller form factor, larger system integration, and field ready designs. The existing designs should prove as cornerstones for engineers that continue the project development.

VII. ACKNOWLEDGEMENT

We would like to express our gratitude to IoT4Ag and Agribugs for their financial support that made this research possible. Special thanks to Dr. David Arnold for providing access to resources essential for this project. Additionally, we would like to thank Trevor Free for his discussions and procurement contributions. Finally, we would like to thank Dr. Christopher Dougherty for his valuable mentorship.

VIII. REFERENCES

- [1] "Analog-to-Digital Converter." Wikipedia, Wikimedia Foundation, Oct. 9, 2023. [Online]. Available: [\[en.wikipedia.org/wiki/Analog-to-digital_converter\]](https://en.wikipedia.org/wiki/Analog-to-digital_converter). Accessed: Dec. 2, 2023.
- [2] Texas Instruments. "BQ24650EVM Synchronous, Switch-Mode, Battery Charge Controller for Solar Power." [Online]. Available: [\[www.ti.com/lit/pdf/sluu444\]](https://www.ti.com/lit/pdf/sluu444). Accessed: Dec. 2, 2023.
- [3] Texas Instruments. "CD4066B CMOS Quad Bilateral Switch Datasheet (Rev. H)." [Online]. Available: [\[www.ti.com/lit/ds/symlink/cd4066b.pdf\]](https://www.ti.com/lit/ds/symlink/cd4066b.pdf). Accessed: Dec. 2, 2023.
- [4] Texas Instruments. "Current Sensing with Different Types of Amplifiers." [Online]. Available: [\[www.ti.com/content/dam/videos/external-videos/4/3816841626001/6076312538001.mp4/subassets/current-sense-amplifiers-current-sensing-with-different-types-of-amplifiers-presentation-quiz.pdf\]](https://www.ti.com/content/dam/videos/external-videos/4/3816841626001/6076312538001.mp4/subassets/current-sense-amplifiers-current-sensing-with-different-types-of-amplifiers-presentation-quiz.pdf). Accessed: Dec. 2, 2023.
- [5] Texas Instruments. "DRV8871 3.6-A Brushed DC Motor Driver with Internal Current." [Online]. Available: [\[www.ti.com/lit/ds/symlink/drv8871.pdf\]](https://www.ti.com/lit/ds/symlink/drv8871.pdf).
- [6] P. Horowitz and W. Hill, "The Art of Electronics." Cambridge University Press, 2022.
- [7] "I2C Communication Protocol." Wikipedia, Wikimedia Foundation, Oct. 3, 2023. [Online]. Available: [\[en.wikipedia.org/wiki/I%C2%BA%C2%B2C\]](https://en.wikipedia.org/wiki/I%C2%BA%C2%B2C). Accessed: Dec. 2, 2023.
- [8] Infineon Technologies. "IRF5210." [Online]. Available: [\[www.infineon.com/cms/en/product/power/mosfet/p-channel/irf5210/\]](https://www.infineon.com/cms/en/product/power/mosfet/p-channel/irf5210/). Accessed: Dec. 2, 2023.
- [9] Texas Instruments. "INA181 Bidirectional, Low- and High-Side Voltage Output, Current-Sense ..." [Online]. Available: [\[www.ti.com/lit/ds/symlink/ina181.pdf\]](https://www.ti.com/lit/ds/symlink/ina181.pdf). Accessed: Dec. 2, 2023.
- [10] Texas Instruments. "LM358 Industry-Standard Dual Operational Amplifiers." [Online]. Available: [\[www.ti.com/lit/ds/symlink/lm358.pdf\]](https://www.ti.com/lit/ds/symlink/lm358.pdf). Accessed: Dec. 2, 2023.
- [11] Texas Instruments. "LM4040 Precision Micropower Shunt Voltage Reference Datasheet (Rev. N)." [Online]. Available: [\[www.ti.com/lit/ds/symlink/lm4040.pdf\]](https://www.ti.com/lit/ds/symlink/lm4040.pdf). Accessed: Dec. 2, 2023.
- [12] M. I. Montrose, "Bypassing and Decoupling," in "Printed Circuit Board Design Techniques for EMC Compliance: A Handbook for Designers," Wiley IEEE Press Imprint, 2000.
- [13] M. Murnane and A. Ghazel, "A Closer Look at State of Charge (SOC) and State of Health (SOH) Estimation Techniques for Batteries," Analog Devices, Mar. 31, 2023. [Online]. Available: [\[www.analog.com/en/technical-articles/a-closer-look-at-state-of-charge-and-state-health-estimation-tech.html\]](https://www.analog.com/en/technical-articles/a-closer-look-at-state-of-charge-and-state-health-estimation-tech.html). Accessed: Dec. 2, 2023.
- [14] "Raspberry Pi Pico Python SDK." Raspberry Pi Foundation, [Online]. Available: [\[datasheets.raspberrypi.com/pico/raspberry-pi-pico-python-sdk.pdf\]](https://datasheets.raspberrypi.com/pico/raspberry-pi-pico-python-sdk.pdf). Accessed: Dec. 2, 2023.
- [15] "Raspberry Pico Documentation." Raspberry Pi Foundation, [Online]. Available: [\[www.raspberrypi.com/documentation/microcontrollers/raspberry-pi-pico.html\]](https://www.raspberrypi.com/documentation/microcontrollers/raspberry-pi-pico.html). Accessed: Dec. 2, 2023.
- [16] Texas Instruments. "TCA9548A Low-Voltage 8-Channel I2C Switch with Reset Datasheet (Rev. G)." [Online]. Available: [\[www.ti.com/lit/ds/symlink/tca9548a.pdf\]](https://www.ti.com/lit/ds/symlink/tca9548a.pdf). Accessed: Dec. 2, 2023.
- [17] ST Microelectronics. "TIP142 Complementary Power Darlington Transistors." [Online]. Available: [\[www.st.com/resource/en/datasheet/tip142.pdf\]](https://www.st.com/resource/en/datasheet/tip142.pdf). Accessed: Dec. 2, 2023.
- [18] Texas Instruments. "TLV27X Family of 550-MA/CH, 3-MHz, Rail-to-Rail Output Operational Amplifiers." [Online]. Available: [\[www.ti.com/lit/ds/symlink/tlv272.pdf\]](https://www.ti.com/lit/ds/symlink/tlv272.pdf). Accessed: Dec. 2, 2023.
- [19] Texas Instruments. "TPS51397A Step-down Converter Evaluation Module User's Guide." [Online]. Available: [\[www.ti.com/lit/ug/sbvu066/sbvu066.pdf?ts=1600197483424&ref_url=https%253A%252F%252Fwww.ti.com%252Ftool%252FTPS51397AE_VM\]](https://www.ti.com/lit/ug/sbvu066/sbvu066.pdf?ts=1600197483424&ref_url=https%253A%252F%252Fwww.ti.com%252Ftool%252FTPS51397AE_VM). Accessed: Dec. 2, 2023.
- [20] Texas Instruments. "TPS55288 36-V, 16-A Buck-Boost Converter with I2C Interface Datasheet." [Online]. Available: [\[www.ti.com/lit/gpn/tps55288\]](https://www.ti.com/lit/gpn/tps55288). Accessed: Dec. 2, 2023.

IX. APPENDIX

A. Schematic and PCB - Dieter

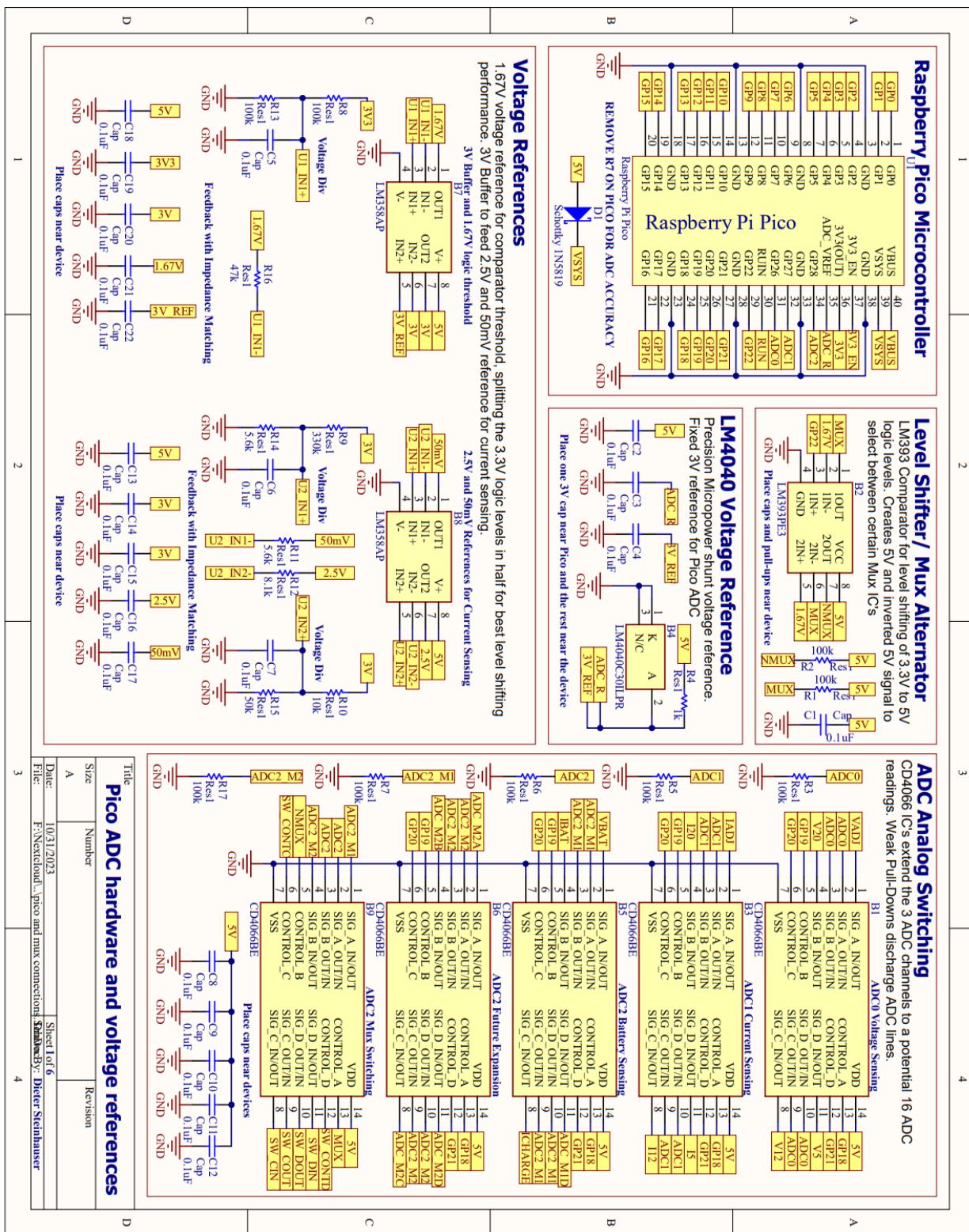


Figure 69: Power Board Schematic Page 1/6.

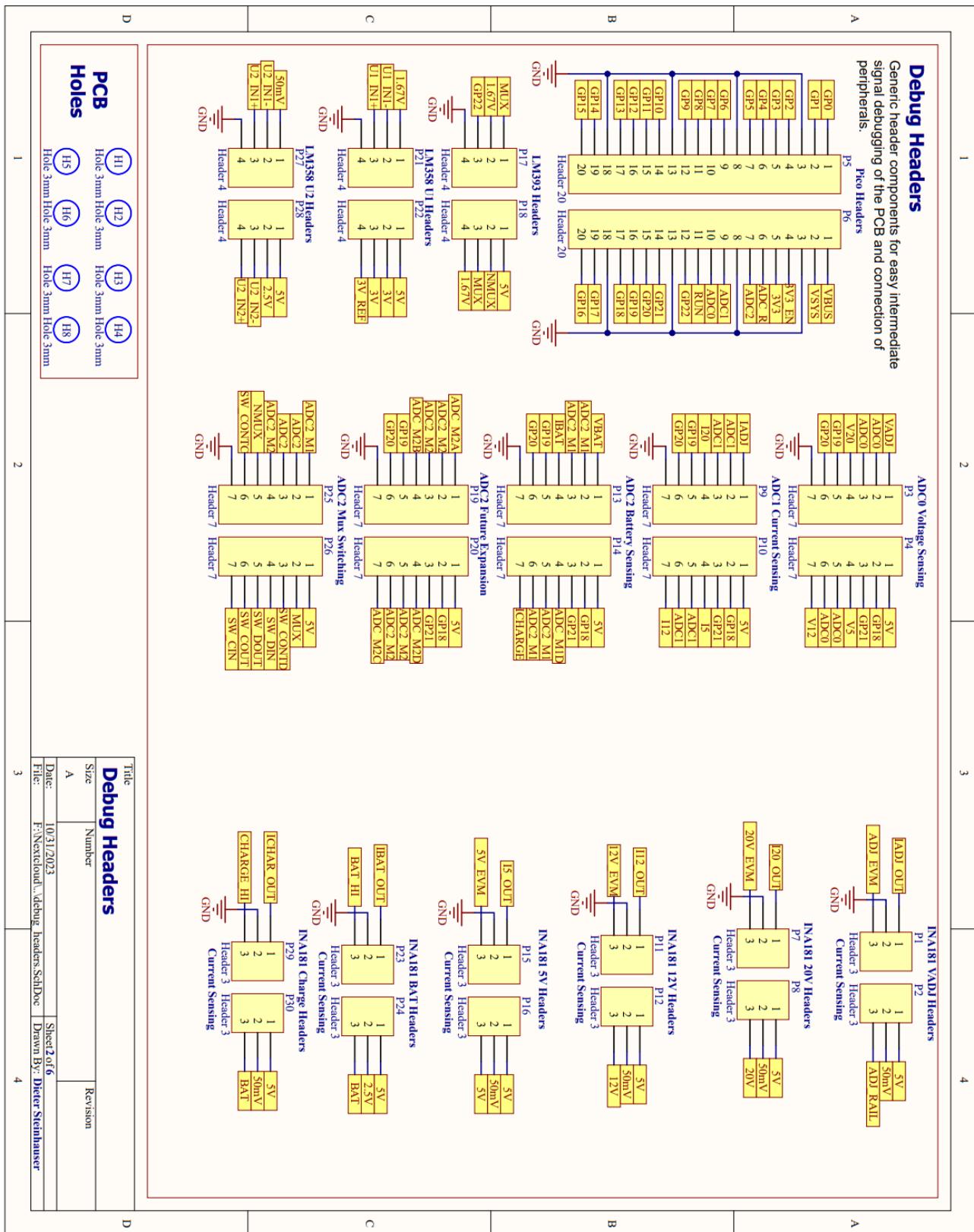
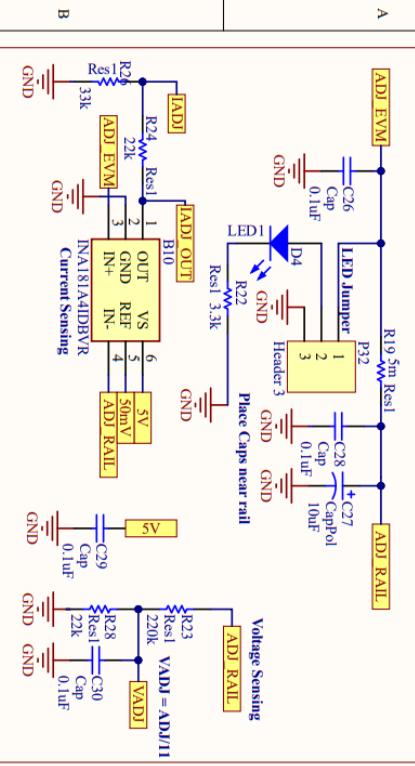


Figure 70: Power Board Schematic Page 2/6.

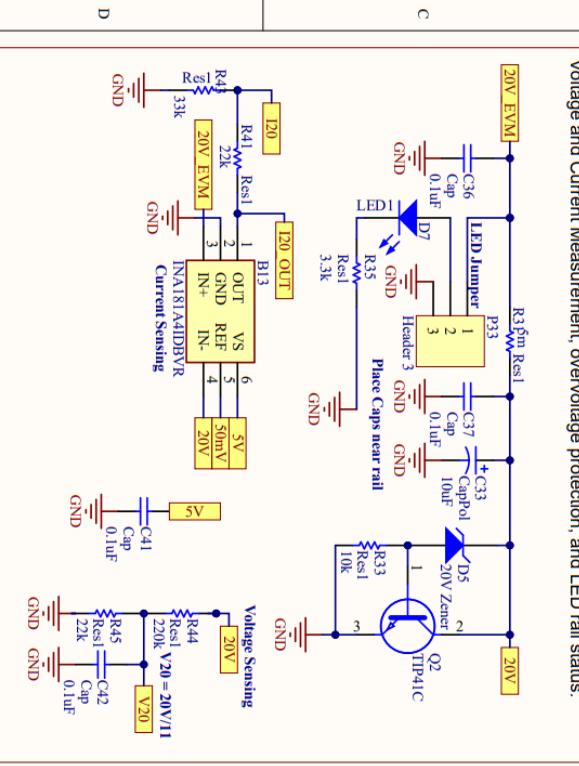
Variable 20V/5A Rail Measurement

Voltage and Current Measurement and LED rail status.



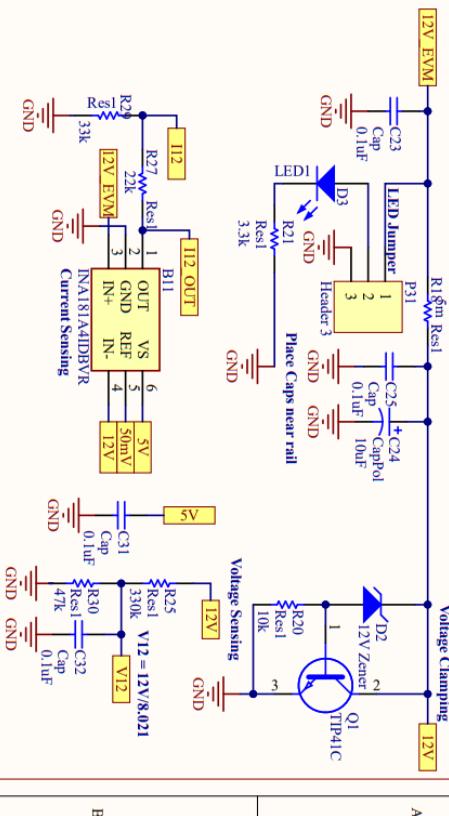
20V/5A Rail Measurement

Voltage and Current Measurement, overvoltage protection, and LED rail status.



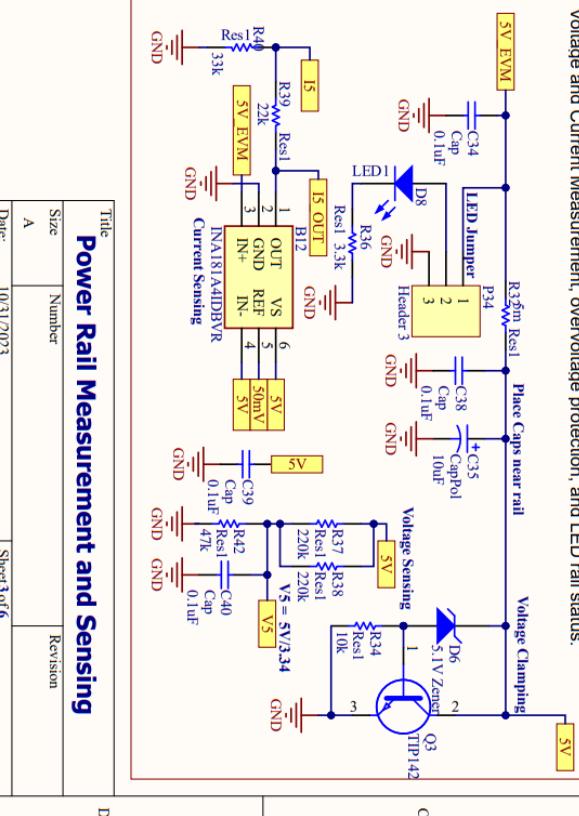
Variable 12V/5A Rail Measurement

Voltage and Current Measurement, overvoltage protection, and LED rail status.



5V/10A Rail Measurement

Voltage and Current Measurement, overvoltage protection, and LED rail status.



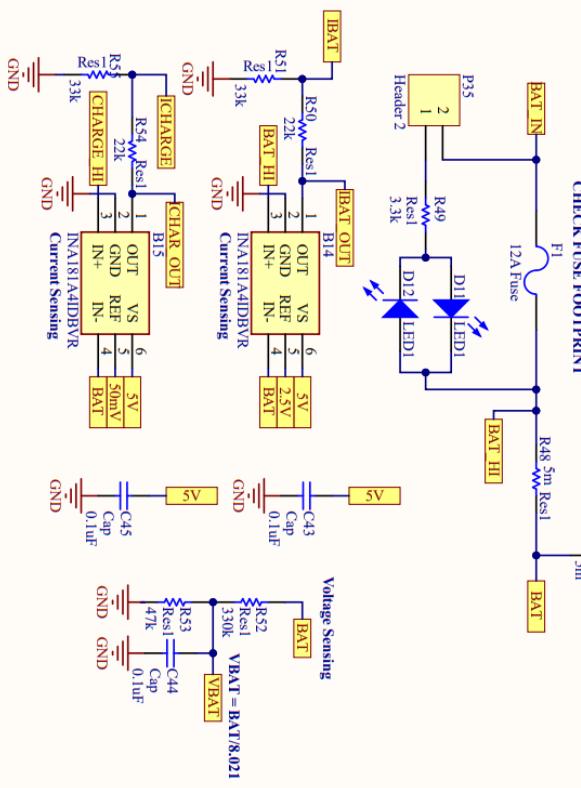
Power Rail Measurement and Sensing

Title		
Size	Number	Revision
A		
Date:	10/31/2023	Sheet 3 of 6
Title:	F:\nextcloud\...\voltage rails Schematic	Drawn By: Dieter Steinhausen

Figure 71: Power Board Schematic Page 3/6.

Battery and Charger Measurement

Battery and Charger connection, Voltage and current measurement. Fuse



System power switch

Latching switch with inline PMOS, supplied with battery voltage.

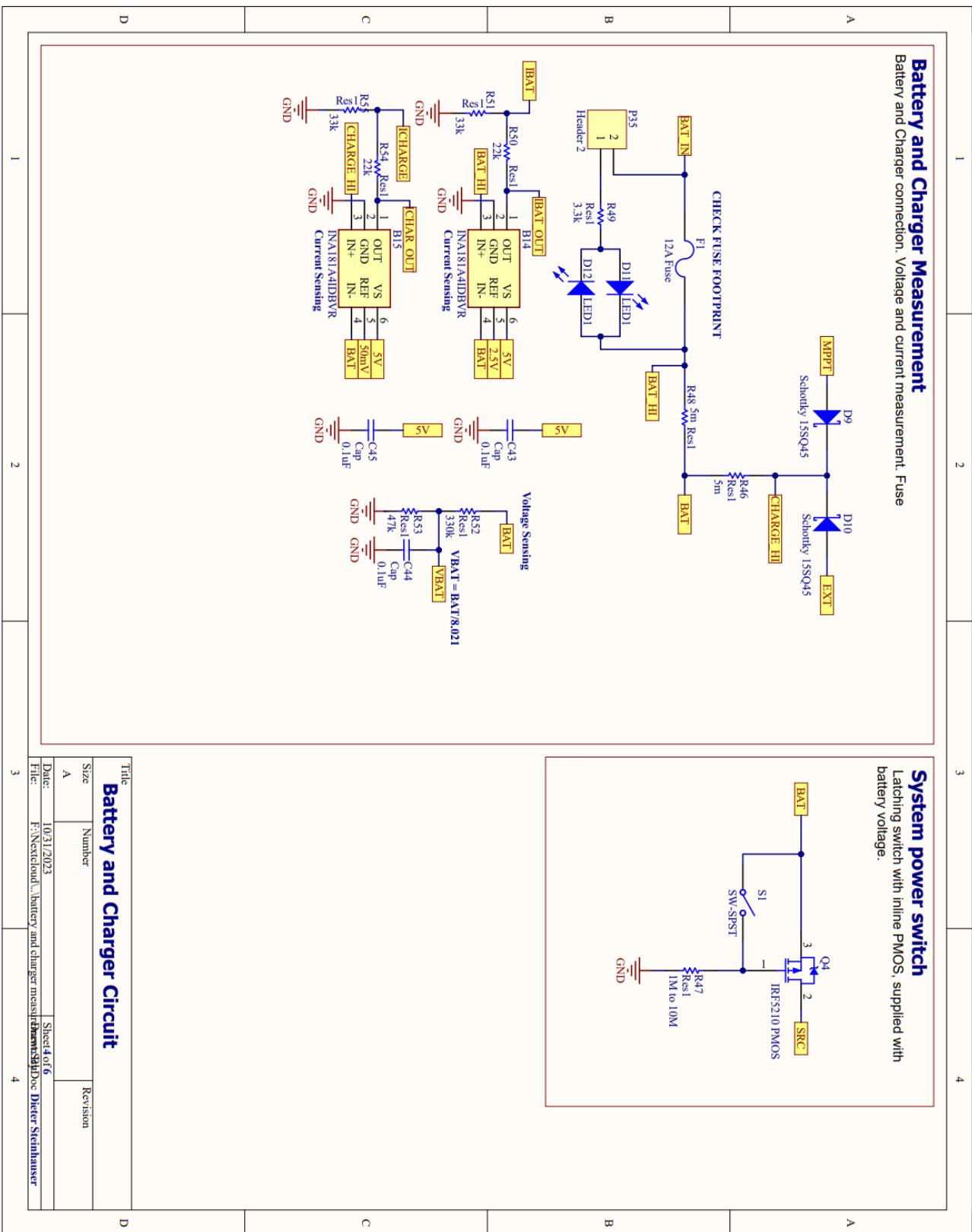
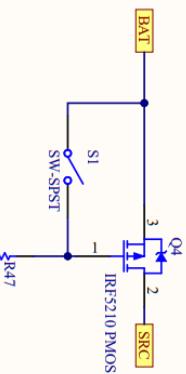


Figure 72: Power Board Schematic Page 4/6.

Battery and Charger Circuit

Title	Size	Number	Revision
Battery and Charger Circuit	A		
Date:	10/31/2023		Sheet 4 of 6
File:	F:\nextcloud\Battery and charger measurement\Dieter Steinhausen		

1 2 3 4

Power Headers

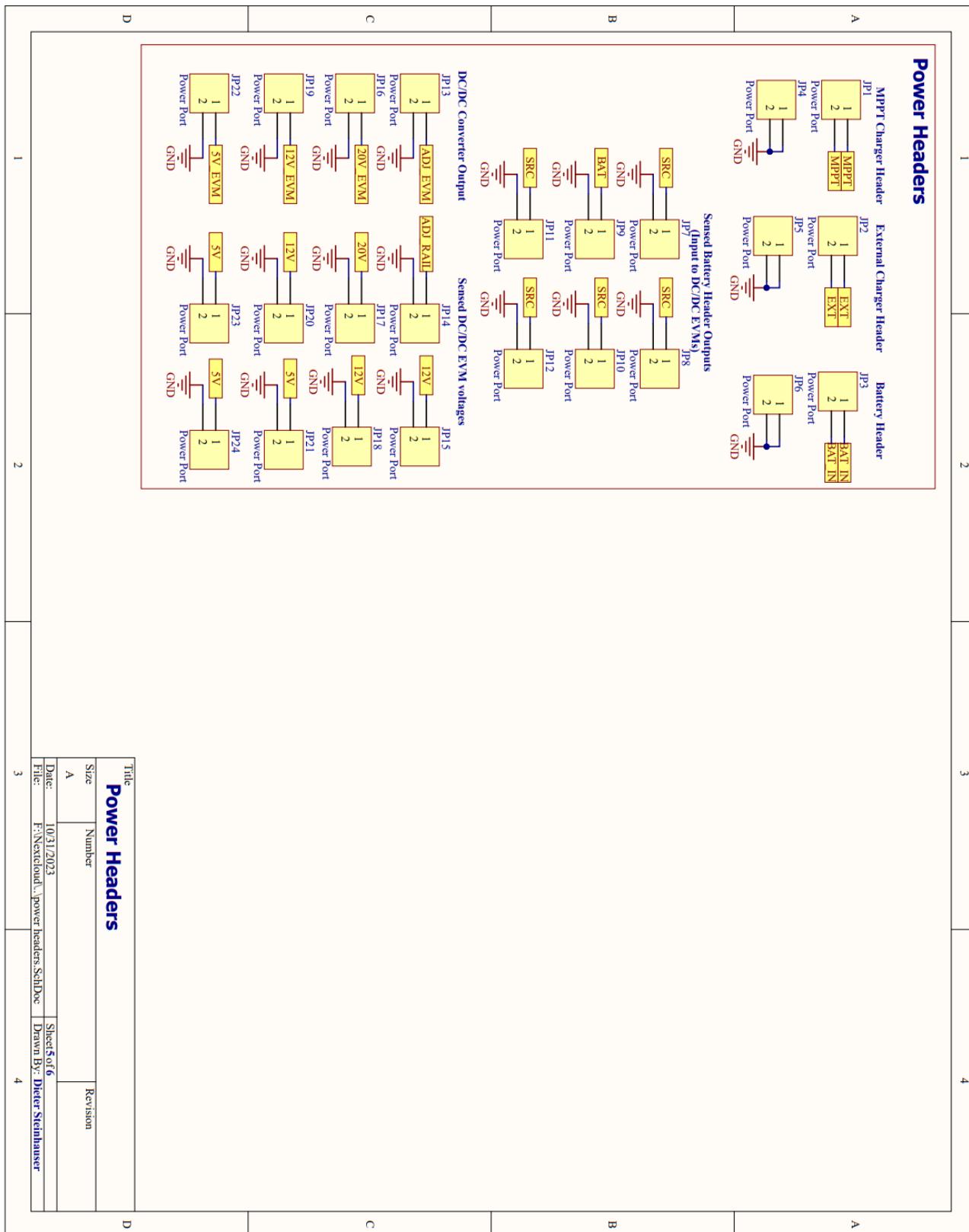


Figure 73: Power Board Schematic Page 5/6.

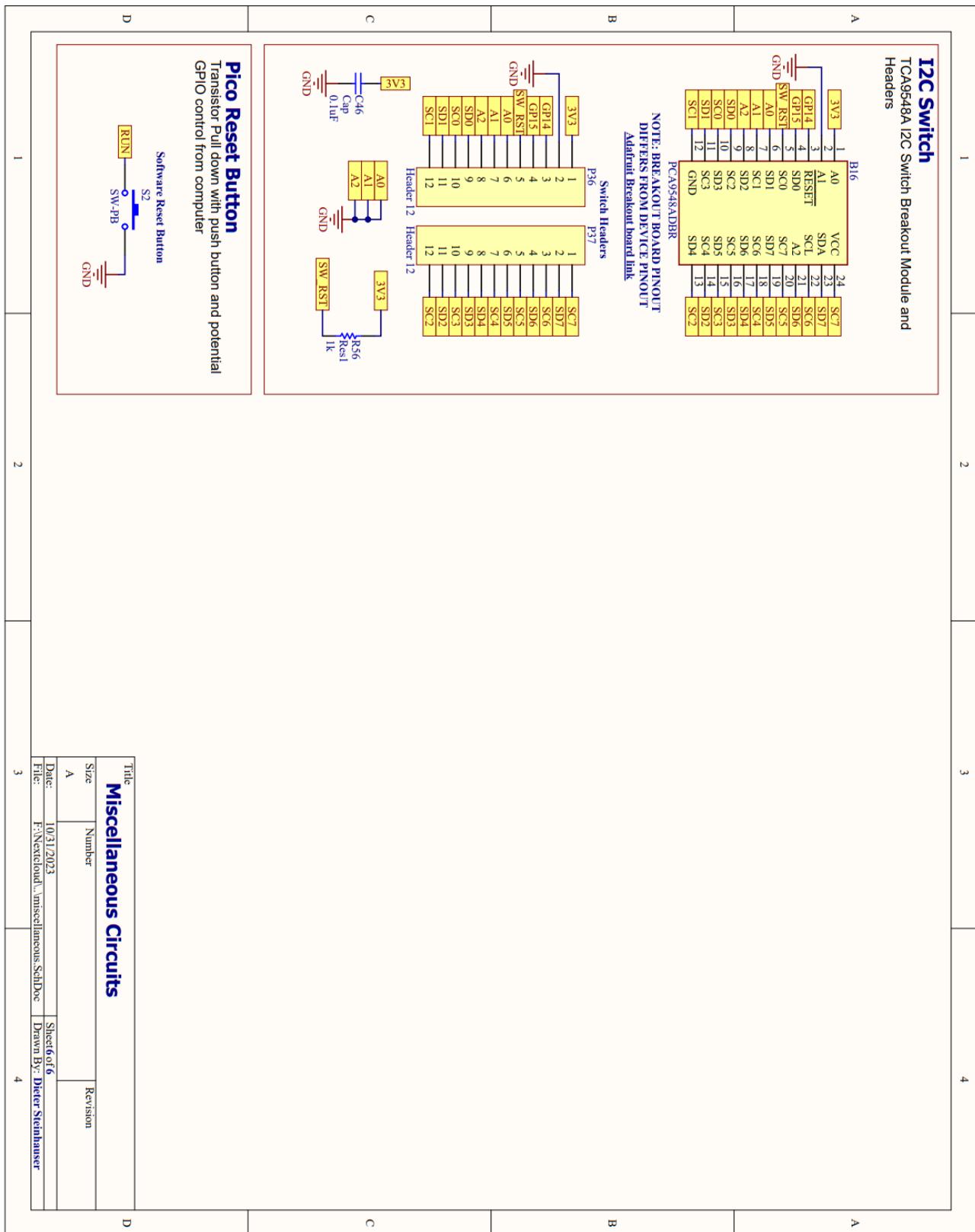


Figure 74: Power Board Schematic Page 6/6.

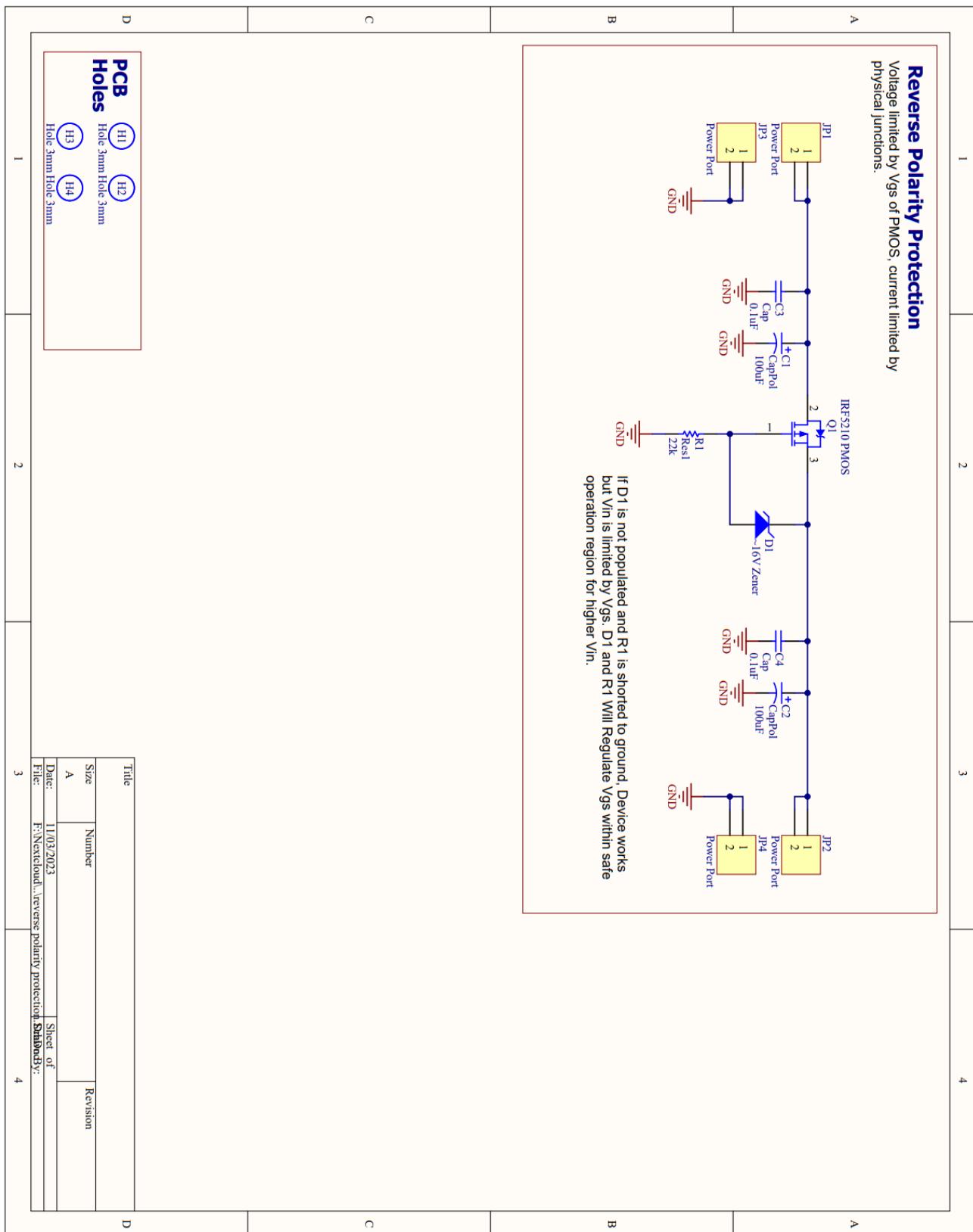


Figure 75: Reverse polarity protection schematic.

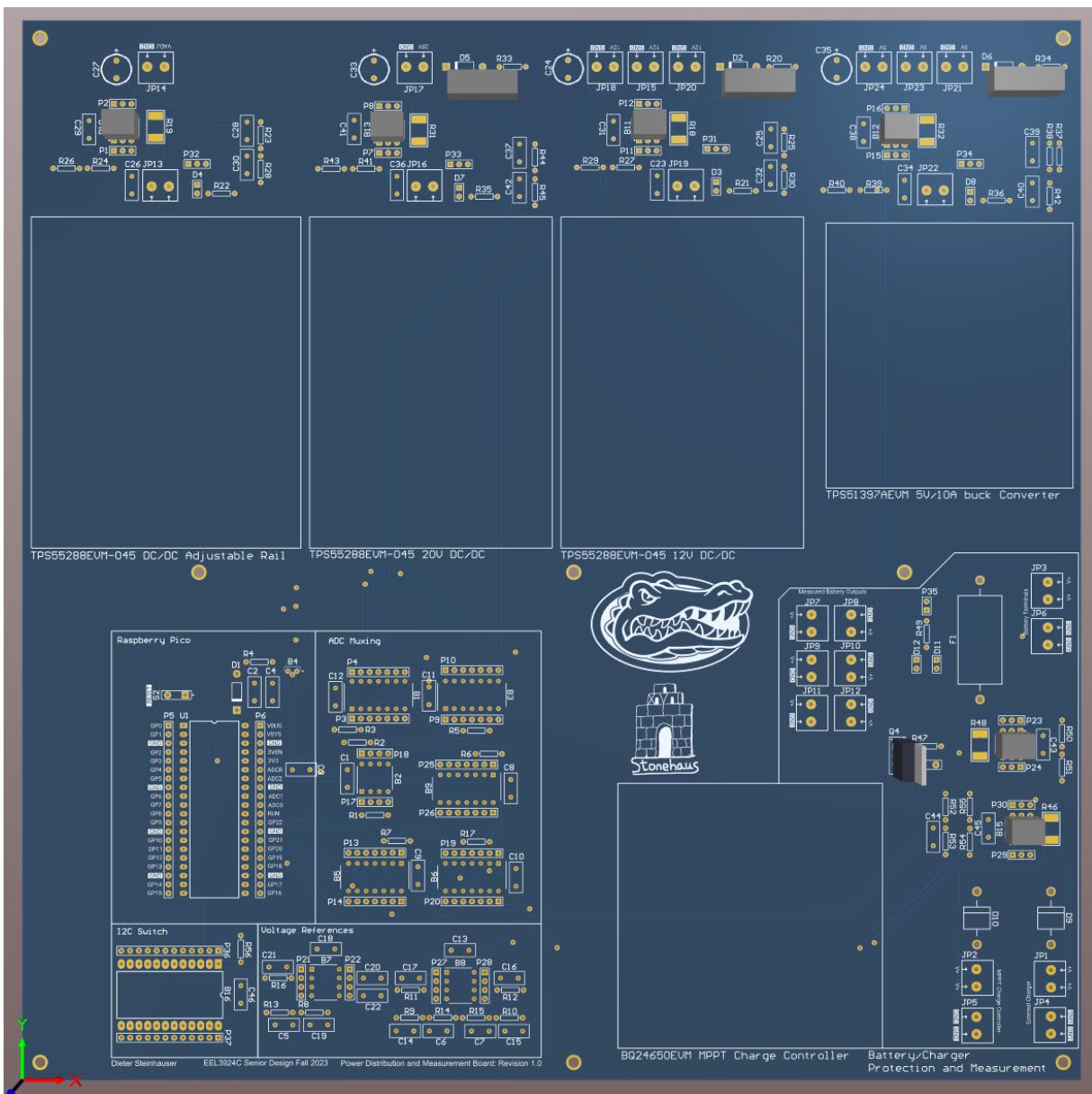


Figure 76: Power Board 3D top view.

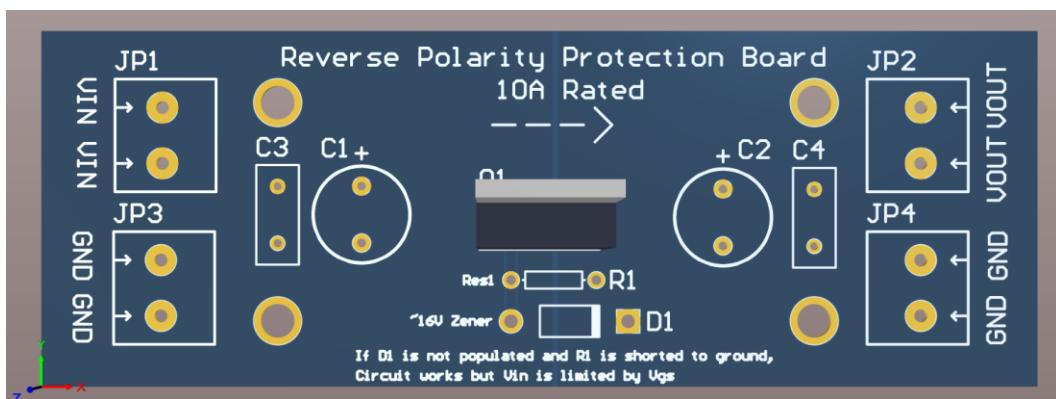


Figure 77: Reverse polarity protection 3D top view.

B. Bill of Materials - Dieter

Name	Description	Designators	Quantity	Price	Component Total	Link
Beelink Series 5	Mini PC running Windows 11	NA	1	\$ 330.00	\$330.00	https://a.co/d/2AIAYeE9
Router	TPlink AC750 Mini Travel Router	NA	1	\$ 40.00	\$40.00	https://a.co/d/2XYZnPO
Battery	Lithium Iron Phosphate 50Ah	NA	1	\$ 140.00	\$140.00	https://a.co/d/3L3fLDB
BBQ24650E VM-639	MPPT Battery Charge controller	NA	1	\$ 100.00	\$100.00	https://www.ti.com/tool/BQ24650EVM-639
TPS55288E VM	20V, 6A, Buck Boost EVM	NA	3	\$ 100.00	\$300.00	https://www.ti.com/product/TPS55288
TPS51387E VM	5V, 10A Buck EVM	NA	1	\$ 50.00	\$50.00	https://www.ti.com/tool/TPS51387AEVM
CD4066BE	CMOS Quad Bilateral Switch	B1, B3, B5, B6, B9	5	\$ 0.51	\$2.55	https://mou.sr/3ehdaZF
LM393PE3	Dual Differential Comparator	B2	1	\$ 0.47	\$0.47	https://mou.sr/47DGS1y
LM4040C30 ILPR	Precision Micropower Shunt Voltage Reference	B4	1	\$ 0.88	\$0.88	https://mou.sr/47UvKxv
LM358AP	Indust Standard Dual Package Operational Amplifier	B7, B8	2	\$ 0.20	\$0.40	https://mou.sr/412XWvp
INA181A4I DBVR	Current Sense Amplifiers 26V, Bidirectional, 350kHz Bandwidth, Low- or High-Side Voltage Output Current Sense Amplifier 6- SOT-23 -40 to 125	B10, B11, B12, B13	4	\$ 0.49	\$1.96	https://mou.sr/3Gpj83
INA181A2I DBVR	Current Sense Amplifiers 26V, Bidirectional, 350kHz Bandwidth, Low- or High-Side Voltage Output Current Sense Amplifier 6- SOT-23 -40 to 125	B14, B15	2	\$ 0.49	\$0.98	https://mou.sr/3T5RHF2
TCA9548	8 Channel I2C Switch Breakout Board	B16	1	\$ 9.62	\$9.62	https://a.co/d/0aN7aoC
100nF Capacitor	MKT Film Capacitor	C1, C2, C3, C4, C5, C6, C7, C8, C9, C10, C11, C12, C13, C14, C15, C16, C17, C18, C19, C20, C21, C22, C23, C25, C26, C28, C29, C30, C31, C32, C34, C36, C37, C38, C39, C40, C41, C42, C43, C44, C45, C46	42	\$ 0.10	\$4.20	https://a.co/d/iDebBl8
100uF Capacitor	Aluminum Electrolytic Capacitors	C24, C27, C33, C35	4	\$ 0.10	\$0.40	https://a.co/d/gsui21
Schottky 1N5819	Schottky Rectifier	D1	1	\$ 0.26	\$0.26	https://mou.sr/4a1bbj
20V Zener	Zener Diode 1N4747	D5	1	\$ 0.32	\$0.32	https://mou.sr/3N9YOIU
16V Zener	Zener Diode 1N4745	NA	4	\$ 0.32	\$1.28	https://mou.sr/47LzWQ4
12V Zener	Zener Diode 1N4742	D2	1	\$ 0.28	\$0.28	https://mou.sr/3NaDoeD
5V Zener	Zener Diode 1N4733	D6	1	\$ 0.21	\$0.21	https://mou.sr/46LJJ42
LEDs	Standard Through hole LEDs	D3, D4, D7, D8, D11, D12	6	\$ 0.12	\$0.72	https://mou.sr/3t1AodG
Schottky 15SQ45	Schottky Rectifier	D9, D10	2	\$ 0.93	\$1.86	https://mou.sr/3NcEdU9
Fuse	Glass Fuse Holder	F1	1	\$ 1.00	\$1.00	https://a.co/d/fIQJECb
Power Port	5mm Power header	JP1, JP2, JP3, JP4, JP5, JP6, JP7, JP8, JP9, JP10, JP11, JP12, JP13, JP14, JP15, JP16, JP17, JP18, JP19, JP20, JP21, JP22, JP23, JP24	32	\$ 0.45	\$14.40	https://mou.sr/3sZRzwa
Header 20	20 Pin Female Header	P5, P6	2	\$ 0.95	\$1.90	https://mou.sr/3TaIprk
Header 12	12 Pin Female Header	P36, P37	2	\$ 0.95	\$1.90	https://mou.sr/3TaIprk
TIP41C	TIP41C, NPN Bipolar Transistor, 6 A 100 V HFE:15 Power, 3-Pin TO-220	Q1, Q2	2	\$ 1.15	\$2.30	https://mou.sr/46GnGP1
TIP142	TIP142, NPN Bipolar Darlington Transistor, 3-Pin TO-220	Q3	1	\$ 1.82	\$1.82	https://mou.sr/41OH6A4
IRF5210 PMOS	P-Channel Power MOSFET	Q4	3	\$ 2.63	\$7.89	https://mou.sr/46BGfUV

Resistors	Standard Through hole Resistors, 1/2W 1%	R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11, R12, R13, R14, R15, R16, R17, R20, R21, R22, R23, R24, R25, R26, R27, R28, R29, R30, R33, R34, R35, R36, R37, R38, R39, R40, R41, R42, R43, R44, R45, R47, R49, R50, R51, R52, R53, R54, R55, R56	60	\$ 0.10	\$6.00	
5mOhm Resistors	Current Shunt resistors 5mOhm	R18, R19, R31, R32, R46, R48	6	\$ 0.69	\$4.14	https://mou.sr/3ta4fk5
SPST Switch	Power Switch, Active open	S1	1	\$ 1.38	\$1.38	https://a.co/d/dZbXpyk
SW-PB	Pushbutton Switch	S2	1	\$ 0.04	\$0.04	https://a.co/d/hJYnQb4
Raspberry Pi Pico	Microcontroller	U1	1	\$ 5.00	\$5.00	https://mou.sr/3LXGxhT
TLV272	Rail-to-Rail Output Op-Amps	NA	2	\$ 1.17	\$2.34	https://mou.sr/3NawXIG
Micro USB cable	USB to Micro USB cable	NA	3	\$ 3.33	\$9.99	https://a.co/d/98G1BKf
DC power cable	DC Power Cable 2.5mm x 5.5mm	NA	2	\$ 5.00	\$10.00	https://a.co/d/i146kUf
Drone Frame	Carbon Fiber drone Frame	NA	1	\$ 23.00	\$23.00	https://a.co/d/co6vkNA
ESP32 CAM	ESP32 uP with Camera	NA	1	\$ 10.00	\$10.00	https://a.co/d/fBJIrdk
DRV8871	DRV8871 H-Bridge Driver Breakout board	NA	2	\$ 11.87	\$23.74	https://a.co/d/gW6MHOe
					Project Total	\$1,112.63

C. GitHub Design and Code Repository

Design and Software files are available at the associated GitHub repository with additional GitHub Pages for sphinx documentation hosting. Sphinx documentation creates HTML pages with autogenerated structures based on code functions and documentation.

The Project Repository is available here: <https://github.com/DieterSteinhauser/Drone-Station>

The Project Documentation is available here: <https://dietersteinhauser.github.io/Drone-Station/index.html>

D. Additional Figures

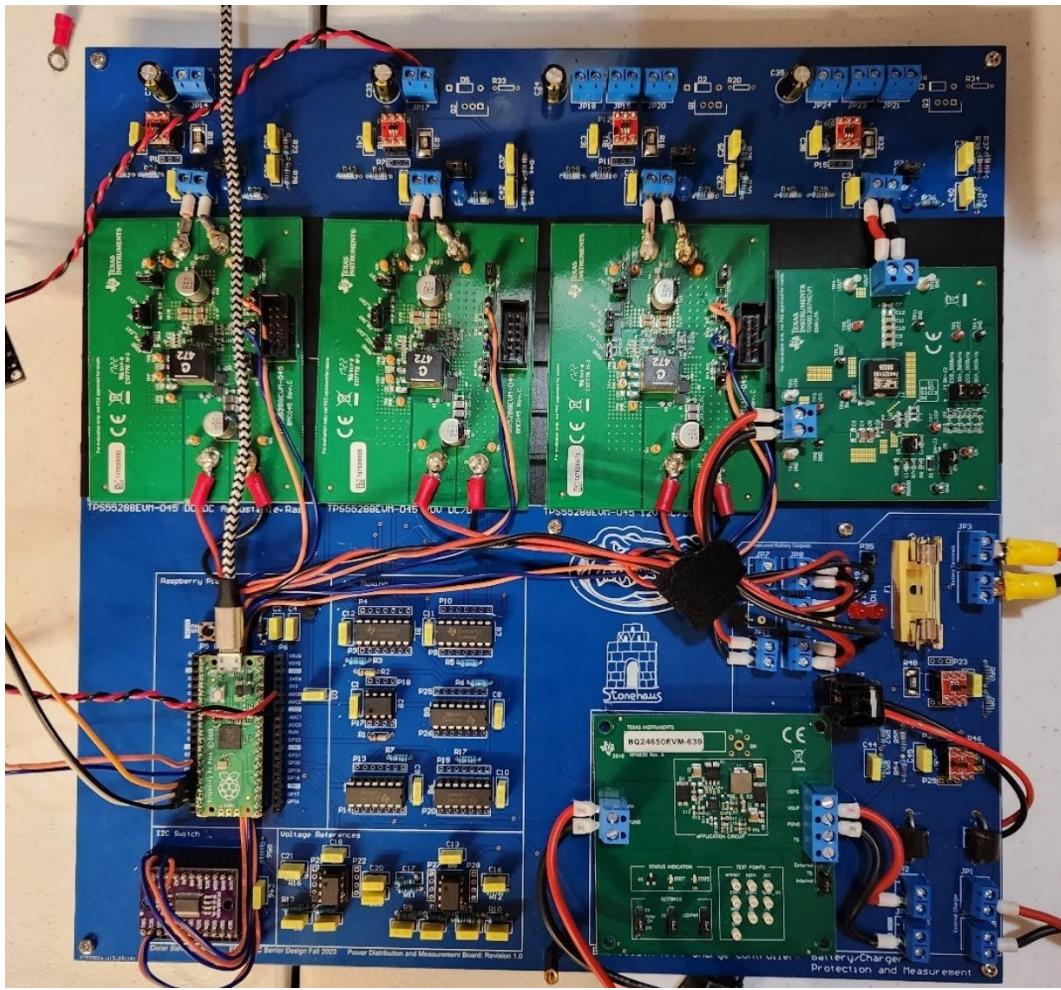


Figure 78: Assembled station power board.

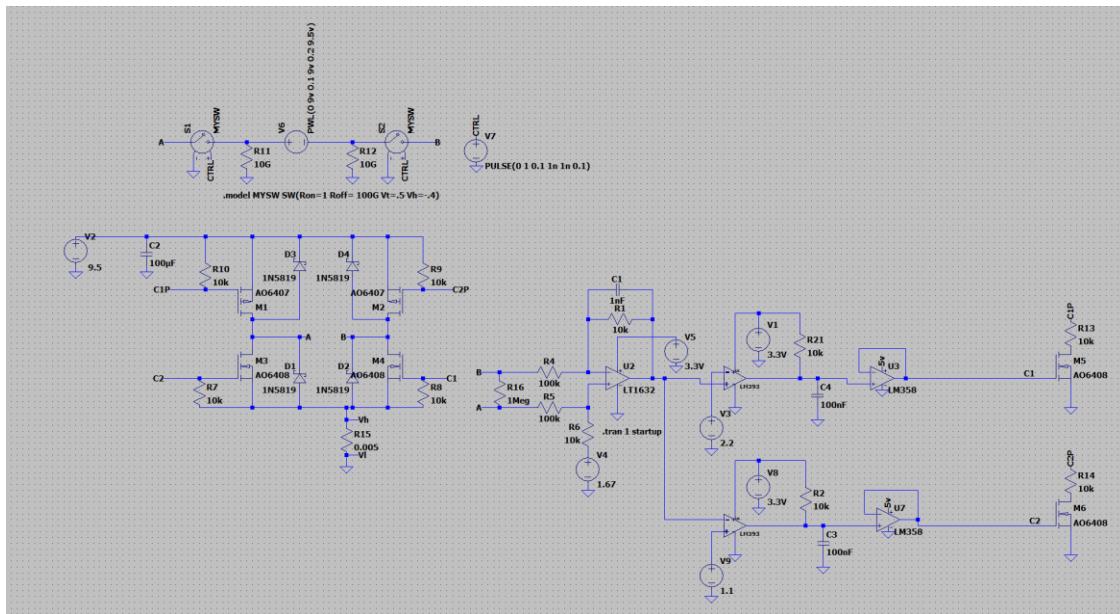


Figure 79: Battery Polarity Detection circuit in LTspice.

E. Omitted Designs

1) Negative Voltage Reference - Dieter

A negative voltage reference is a voltage point below the ground reference. A voltage reference below ground can prove useful for circuits that require approaching or passing the ground rail. In the case of bidirectional current sensing such as monitoring a battery, it may make sense to incorporate a negative reference so that the directionality of current is reflected in the output of a current amplifier. For small load applications such as measurement IC's, a negative voltage reference can be made with a charge pump. A square wave of sufficient frequency can accumulate charge upon capacitors by forcing direction of current using diodes. Efficiency of the system is limited by the frequency matching the flyback capacitor and any losses over the diodes. Energy in the large storage diode can then be regulated in some way or used directly.

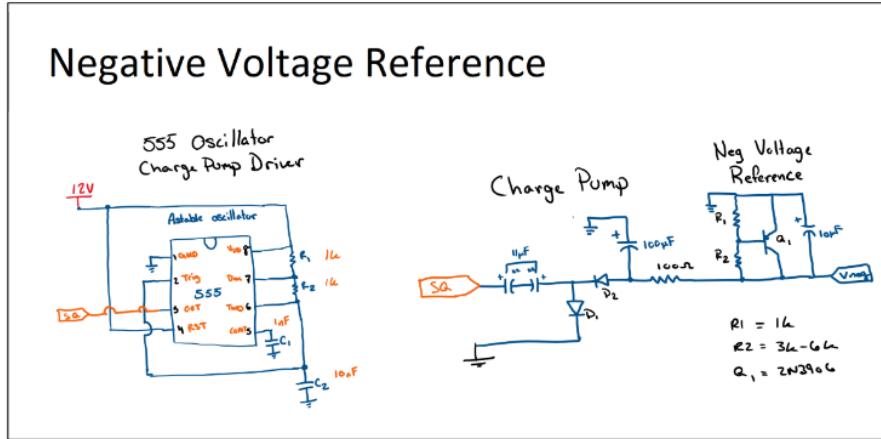


Figure 80: Negative Voltage Reference Circuit. Employs a Square wave generator, charge pump, and Vbe multiplier.

Given a 12V square wave of approximately 40kHz, I was able to achieve voltage of approximately -10V at the output of the charge pump. I chose to regulate the reference using a Vbe multiplier. This exploits the approximate -0.7V drop from base to emitter on a BJT, making the output V_{neg} approximately equal to a ratio of V_{be} and resistance. Since R_1 is 1k ohm, for every additional 1k of resistance seen at R_2 , the output voltage is lowered approximately ~0.7V. Having -0.7V multiplied by R_1 of 1k and R_2 of 6k creates an approximate -5V reference. This method is more complicated than perhaps a Zener diode but uses less quiescent current and introduces less noise into the circuit.



Figure 81: -5.08V Voltage Reference with 10mA load seen on channel 1. The charge pump and 10mA load required 21mA for operation.



Figure 82: Current supplied to the charge pump circuit.

We implemented the circuit and found that the -5V reference could supply a 10mA load comfortably. However, current consumption of the circuit shows that there is about a 50% efficiency, primarily due to the 555-timer frequency and charge pump. The circuit would also add switching noise to the overall system that would be best to avoid. We also experimented with lowering the input square wave to 5V and were able to create a -1V reference that could supply ~4mA. Due to the low efficiency, switching noise of the circuit, circuit footprint, and overall current draw, we decided to abandon the idea of using a negative voltage reference and implemented alternative design choices. A charge pump IC could be used in the future if the need arises as an IC will be more efficient and have a significantly smaller footprint and noise level.