Solar Integrated Battery Bank

Senior Design Team sddec22-02.
A project for PowerFilm Solar

Carter McCarthy, John Fecht, Will Thorne, Nathan Harder, Jordan Hoosman
Advised by: Cheng Huang - chengh@iastate.edu | Overseen by: Dan Stieler - dstieler@powerfilmsolar.com

Our Mission:

Redesign the LightSaver Max, to modernize the design.
Provide our findings and recommendations for the next iteration of the LightSaver to PowerFilm.

Base Product:
LightSaver Max

Original Design

Specifications

- A premium battery bank with a flexible solar panel attached
  2 input ports, and 3 output ports (5 total)
- 2 printed circuit boards (PCB)
- Fixed point power tracking
- 60Wh battery capacity
- 2 end caps / ends with ports

Original Design

Use and Users

- Light weight, Portable, Durable
- Advertised to help charge mobile devices in a pinch, specifically branded towards backpackers and survivalists

Requested Improvements

Changes

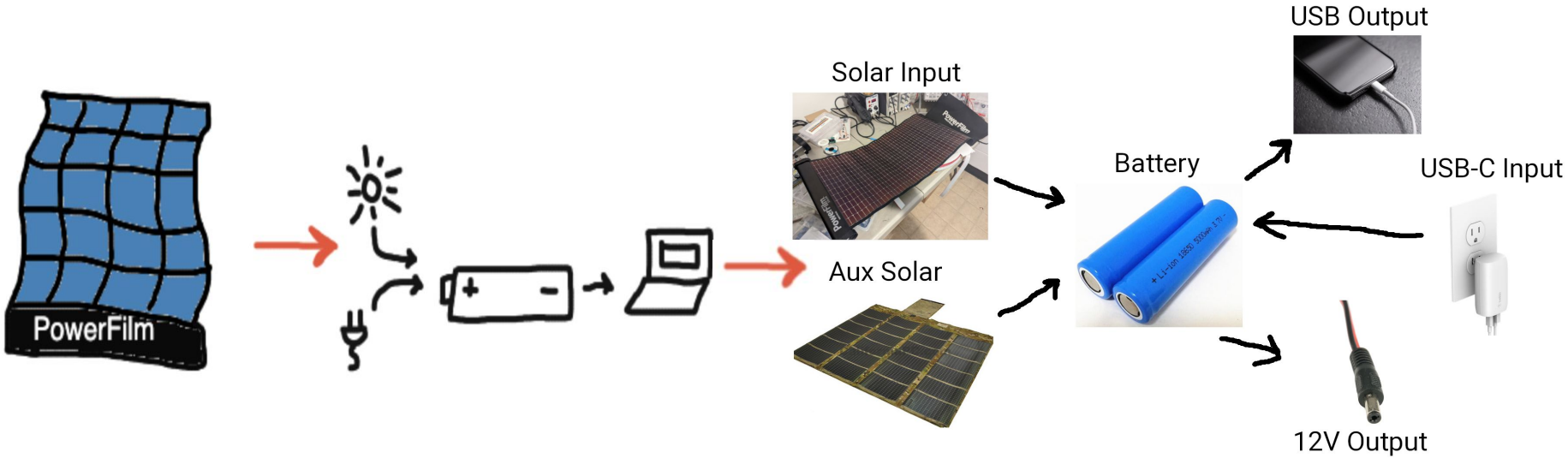
- 3 input ports, and 3 output ports (4 total by use of 2 bi-directional USB-C’s)
- 1 printed circuit board (PCB)
- Maximum power point tracking (MPPT)
- 85Wh battery capacity
- 1 endcap / end with ports

Changes - side by side

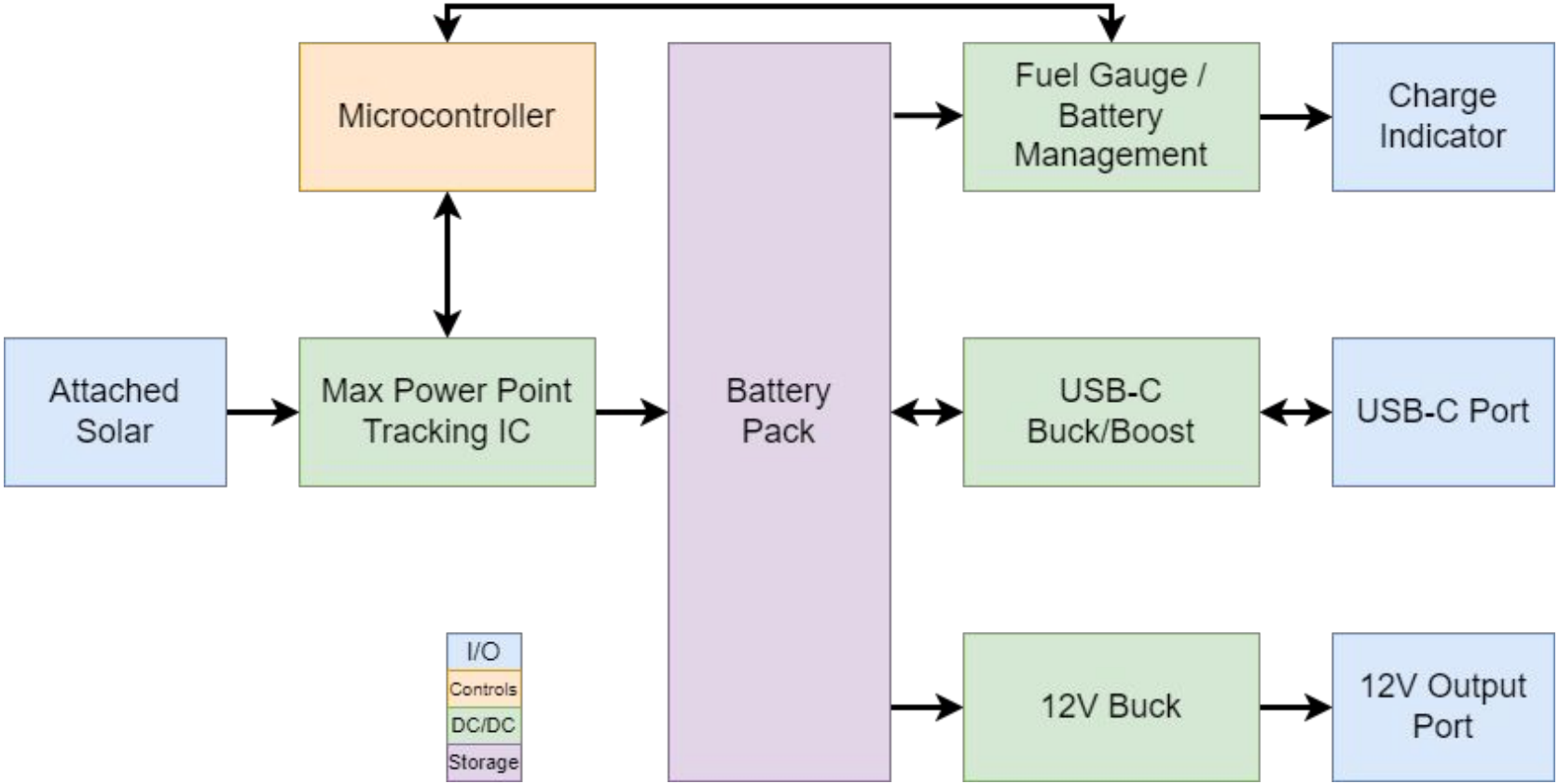
Previous Design	**Future Design**
2 input ports, and 3 output ports (5 total)	3 input ports, and 3 output ports (4 total by use of 2 bi-directional USB-C’s)
2 printed circuit boards (PCB)	1 printed circuit board (PCB)
Fixed point power tracking	Maximum power point tracking (MPPT)
60Wh battery capacity	85Wh battery capacity
2 endcaps / ends with ports	1 endcap / end with ports

Design Plans

Conceptual Diagram



Block Diagram



What and Why

Batteries : 4s2p, li-ion

- Increased battery life
- Fast charging

MPPT : custom design, fabricated

- Increased efficiency of the solar panel

12V Buck/Boost : custom design

- Increased voltage output for loss compensation

USB-C PD : tps25750 (PD management), bq25792 (buck/boost)

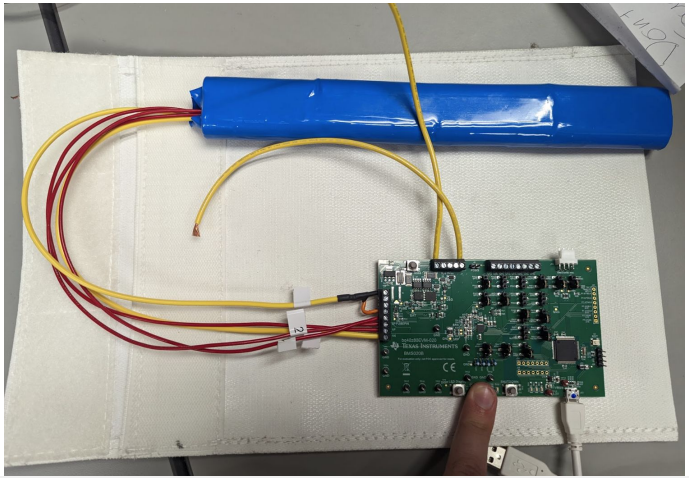
- Standalone usb handshake

Microcontroller : nrf52832

- Low-power, convenient DMA communication options

Implementation

Battery Implementation



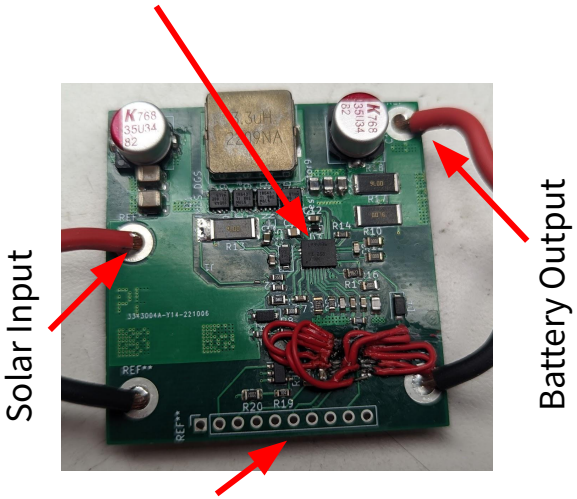
Improvements

- Increased battery size and configuration
- Created battery management systems
- Provided fast charge option
- Designed compatible and tileable battery design

Testing

- Used evaluation board for testing purposes
- Utilized learning cycle to fine tune parameters
- Oscilloscope measurements
- Integration testing

MPPT Implementation

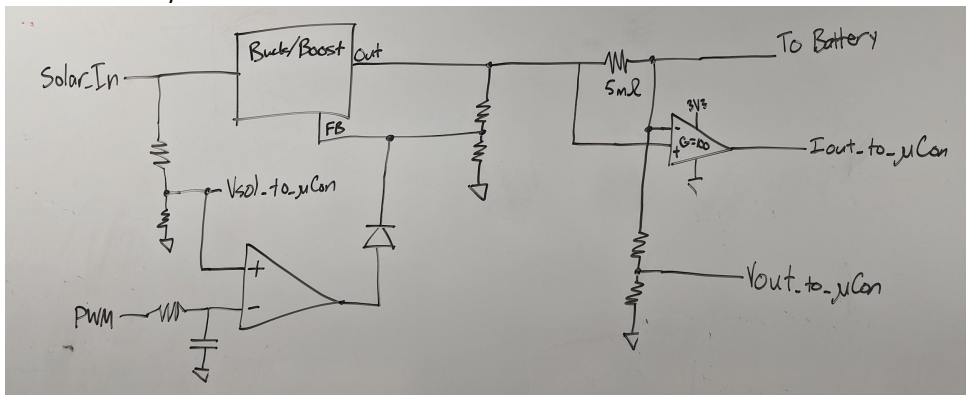


Implementation:

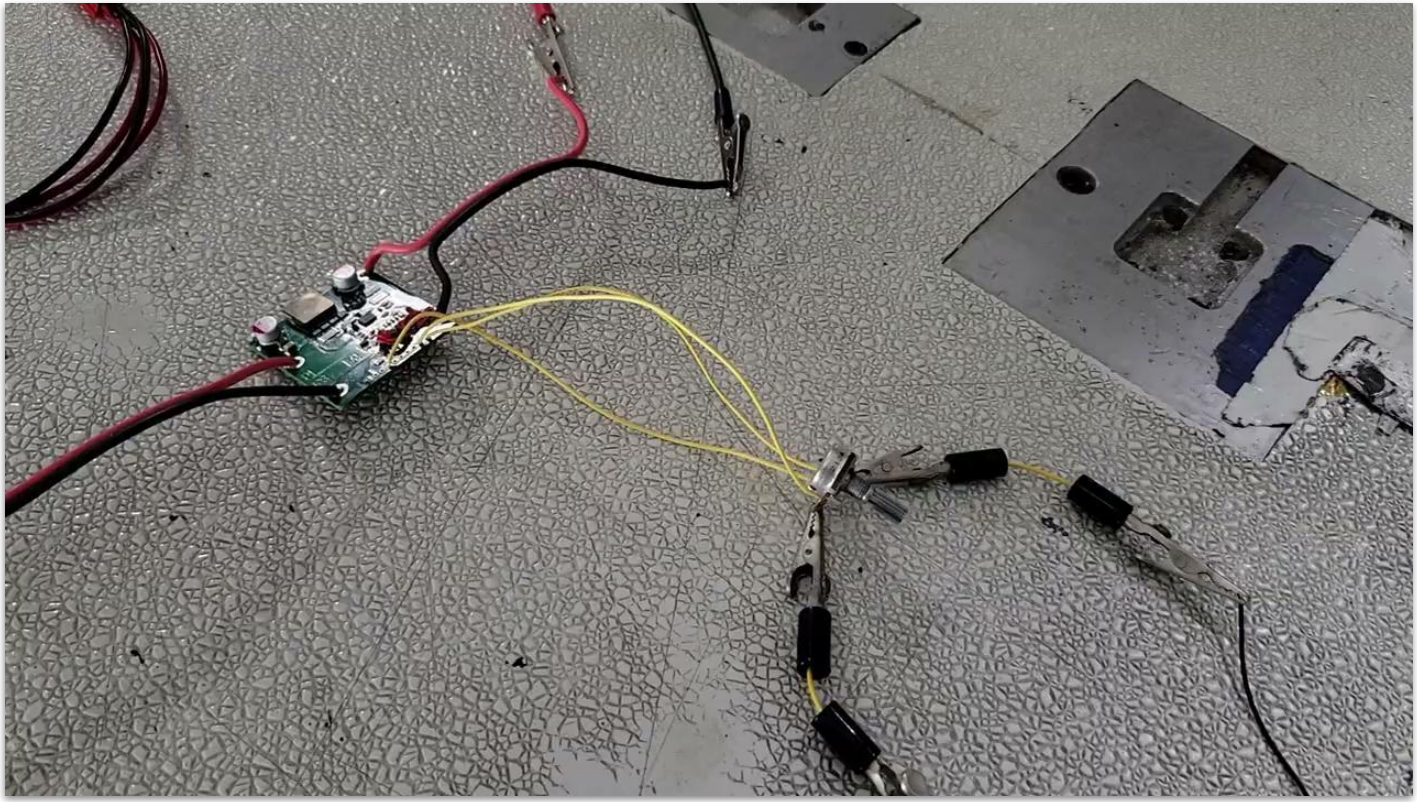
- Custom PCB Layout
- Buck/boost converter with altered feedback loop
- Vsolar, Vout, and Iout are all read from uController
- MPPT algorithm is used to determine MPP

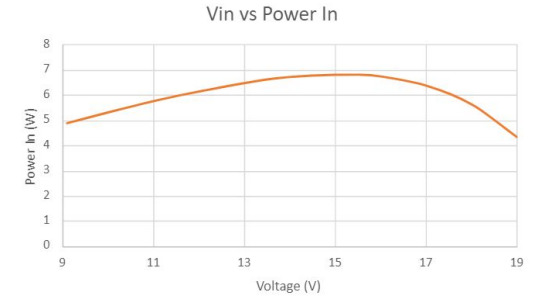
Testing:

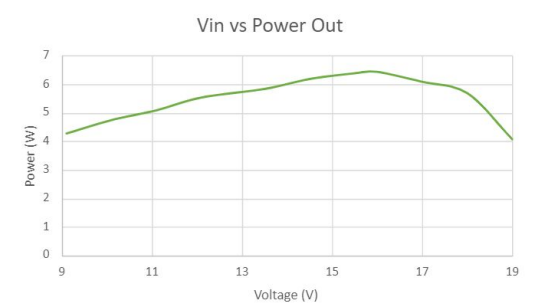
- Use panels with previously tested MPP
- Load tester set to constant current
- Run the algorithm with the uController attached, see if it brings the attached solar to V_MPP



MPPT w/out uController







MPPT Implementation Continued

Results:

- MPPT was able to hold a stable set point voltage
- Output power increases as panel approaches MPP

Issues:

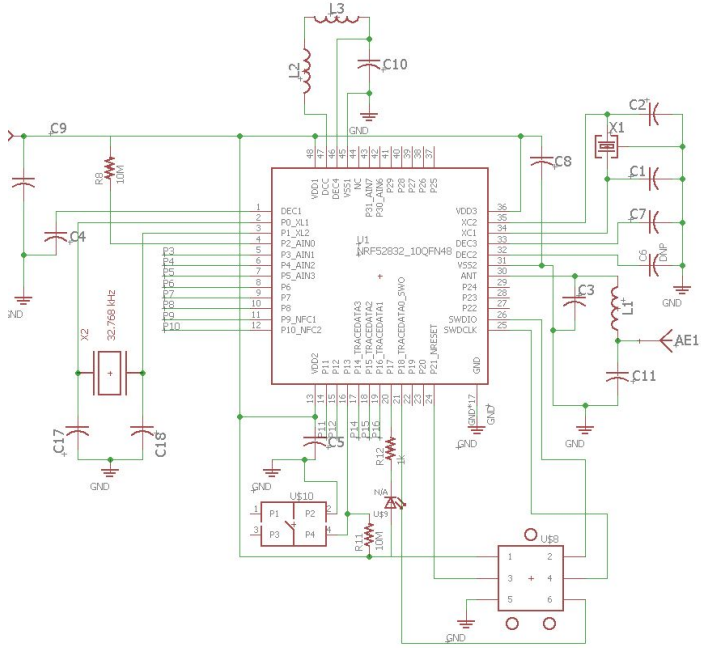
- Signals going to the microcontroller are extremely noisy
- Attempts to resolve:
  - LPF on signals going to microcontroller
  - Move all signal grounds to AGND from PGND
  - Increase input and output capacitance

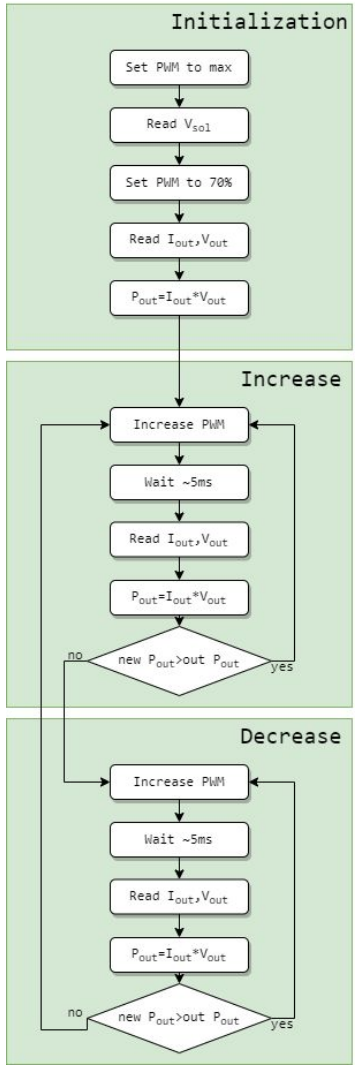
Top and center: Input voltage vs Power in and Power out of MPPT
Bottom: Signal noise observed on Iout line (Blue is prefiltered, yellow is post filtering)

Microcontroller Implementation

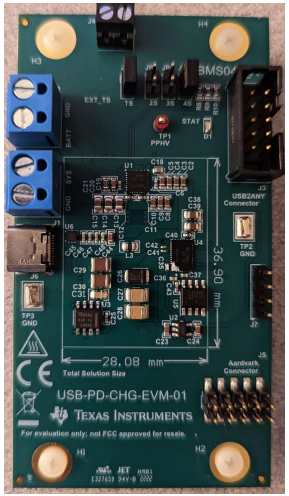
- Using I2C for battery/fuel gauge and PWM for MPPT
- MPPT algorithm implemented in software

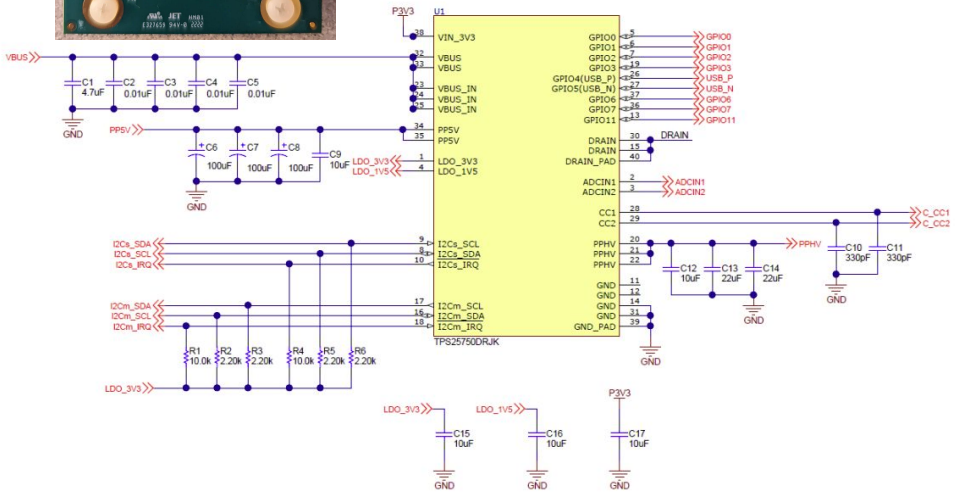
```
int  MPPT_Calculate_Power() {
    int iout = analogRead(MPPT_PIN_IOUT);
    int vout = analogRead(MPPT_PIN_VOUT);
    return iout * vout;
}
void MPPT_Init() {
    // Setup pins
    pinMode(MPPT_PIN_VOUT, INPUT);
    pinMode(MPPT_PIN_IOUT, INPUT);
    pinMode(MPPT_PIN_VSOLAR, INPUT);
    pinMode(MPPT_PIN_PWM, OUTPUT);

    // Set PWM to max
    analogWrite(MPPT_PIN_PWM, 12);
```





USB-C Implementation





TPS25750 - USB-PD Controller

- Handles the USB handshake

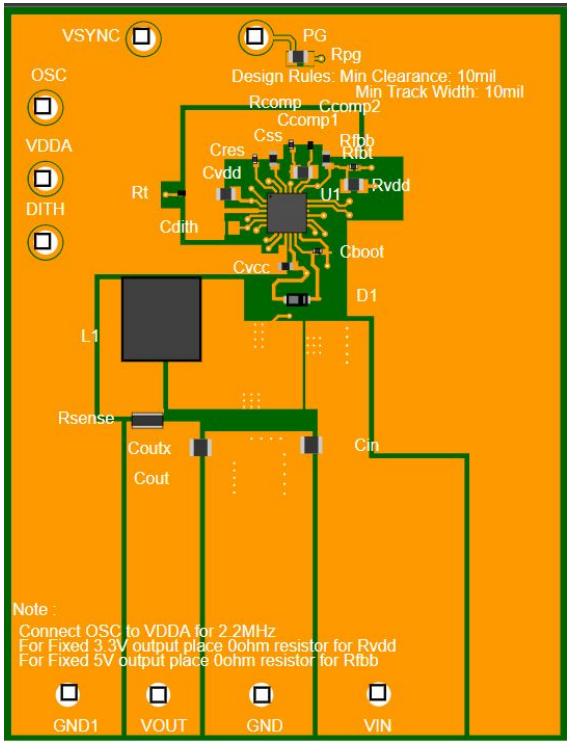
  Including USB-C specification

- Standalone, doesn't need firmware dev. or uController intervention
- Directly interacts with BQ25792 Buck/Boost

Learning Points

- Shipping Issues
  - Try, try, try again
- Implementation Troubles

12V Buck Intended Implementation

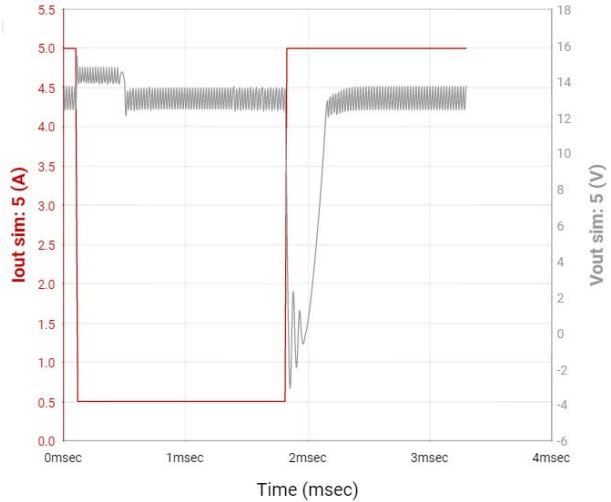


Improvements

- Increased current output

Issues

- PCB fabricated May 2022 but never arrived
- Previous design was for 12V output but is now 13V



Load Transient

12V Buck Intended Implementation Cont.

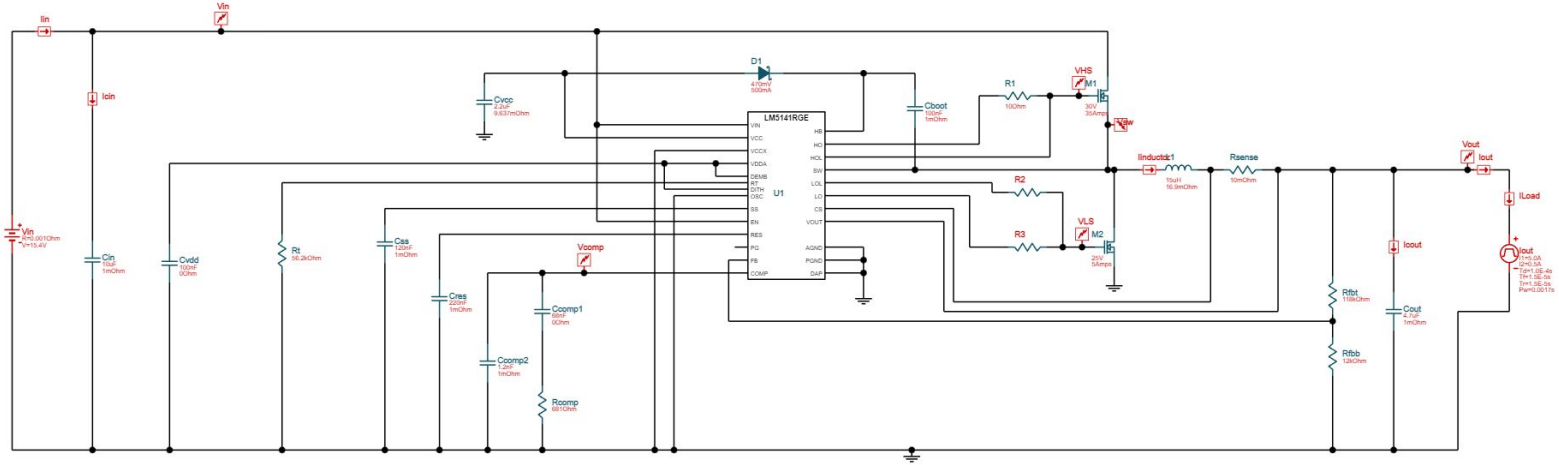
How I would have…

Implemented

- Connected the grounds and Vin pins to the battery regulator

Tested

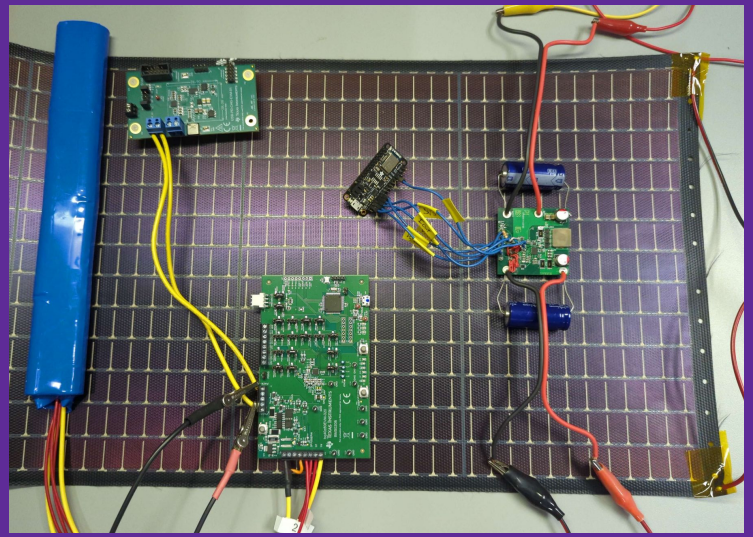
- **Vary the input voltage between 14V and 16.8V while observing the output voltage and current to debug and determine efficiency**



USB-C Testing Board

- USB-C-PD-DUO-EVM
  - A test board created for the sole purpose of testing source and sink through usb

Final Product



Completion Checklist (a list of all the things we finished)

- Battery pack
    - 4s2p battery pack assembled
    - Battery management IC
- MPPT
    - Algorithm implemented and flashed to microcontroller
    - Circuit designed, tested, modified
    - Interconnect tested and working
- USB-C PD
    - USB-C Handshake and power delivery successfully implemented
- 12V Buck/Boost
    - Designed
    - Simulated
    - Part did not arrive

Contributions / Team Management

Members - Roles:

John Fecht - Team Leader and MPPT

Nathan Harder - DC/DC Converter and debugging

Carter McCarthy - battery and charge indicator

Jordan Hoosman - Micro controller

Will Thorne - Digital Communication Protocols and USB-C

Thank You!

Questions?