

## STM32 Projects & Blog Posts

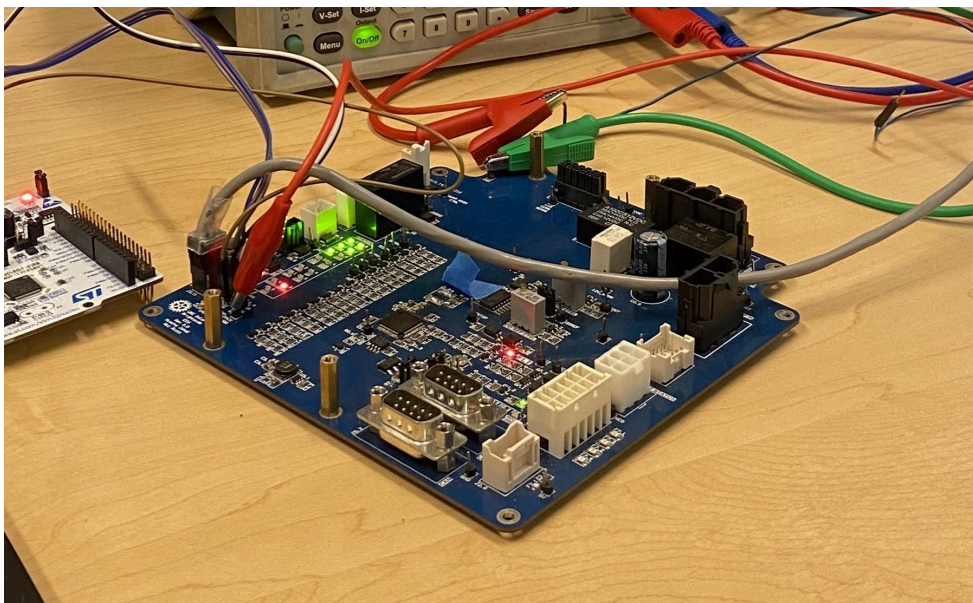
IWDG & RCC\_CSR Blog post:

<https://ckalitin.github.io/technical/2025/03/26/stm32-rcc-register.html>

Automated ADC Characterization:

<https://ckalitin.github.io/projects/2024/12/29/scpi-auto-characterization.html>

In 2024 I joined the UBC Solar student design team on the Battery Management System subteam. Most of my work involved programming, and characterizing STM32 microcontrollers. This involved debugging Independent Watchdog errors (which required diving deep into how STM32s really work), characterizing the ADCs, and more development/programming tasks.



## Python Data Analysis & Simulation Projects

McDowell Dataset Analysis: <https://github.com/CKalitin/mcdowell-dataset-analysis>

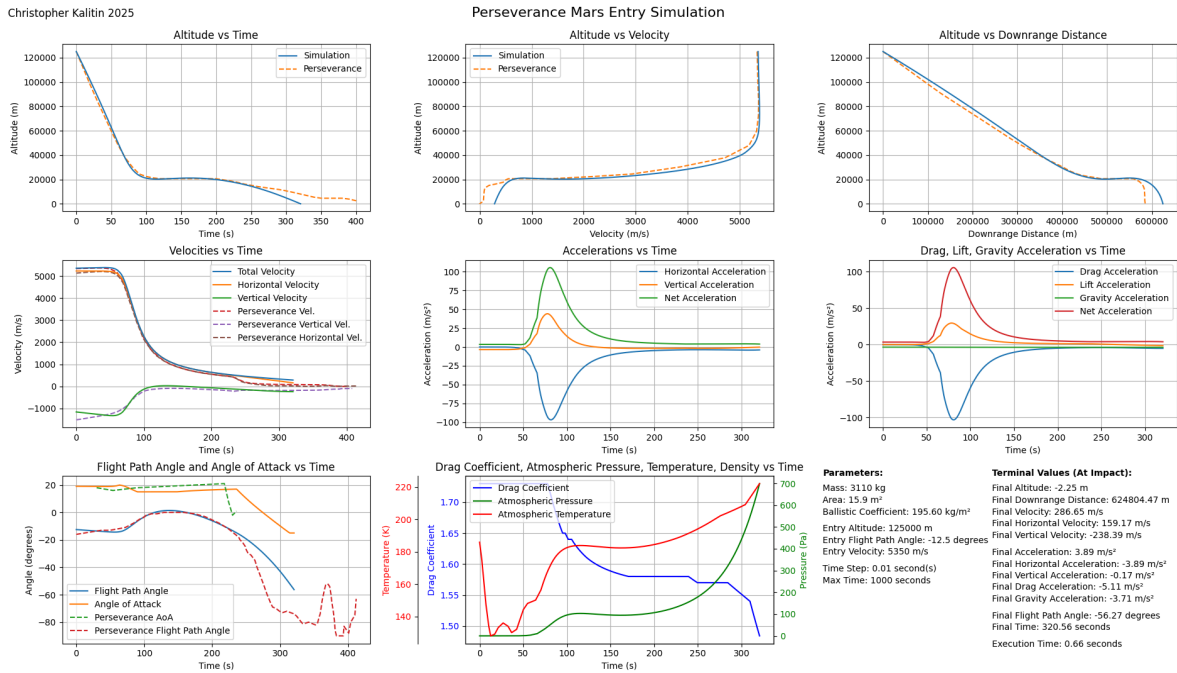
Martian Ballistic Modelling:

<https://ckalitin.github.io/space/2025/05/13/martian-atmosphere-model.html>

I wrote a python library (that's on [PIP!](https://pypi.org/project/ckalitin/)) to analyze Jonathan McDowell's dataset of every launch that's ever occurred and every object to ever be intentionally put into space. He gives raw .tsv files and I wrote a more user friendly Python package.

I also wrote a simulation of blunt body entry vehicles descending through Mars' atmosphere:

Christopher Kalitin



## Analysis of the Commercial Satellite Launch Market

<https://ckalitin.github.io/space/2024/08/12/extrapolating-demand-firefly.html>

[https://x.com/Peter\\_J\\_Beck/status/1823128548581994792](https://x.com/Peter_J_Beck/status/1823128548581994792)

As part of my demand analysis of the commercial small satellite launch market I wrote 10+ blog posts. The most successful of these was read by Peter Beck, the CEO of Rocket Lab!

### Extrapolating Demand and Competition for the 1-ton Rocket Class

Aug 12, 2024 • Christopher Kalitin



Tell me where I'm wrong or just give compliments [here](#).

Today four of the best space investing Youtubers posted their interview with Peter Beck and Adam Spice of Rocket Lab. Responding to one of Dave G's questions, Peter Beck said this about the 1-ton rocket class: "Our view of 1-ton is it's kind of a no man's land."

Christopher Kalitin @CKalitin · Aug 12

After hearing to @Peter\_J\_Beck's responses to @daveinvesting's questions in their recent interview, I wrote this blog post on Alpha's TAM in launch and the lack of overlap with Electron.

"Our view of 1-ton is it's kind of a no mans land" - Peter Beck

[ckalitin.github.io](https://ckalitin.github.io)  
Extrapolating Demand and Competition for the 1-ton

6 6 91 8.3K

Peter Beck @Peter\_J\_Beck

Very well written article. The only other thing to consider is actual payload lift performance vs stated. I have seen that this is often not consistent and you find folks end up lifting loads way lower than their stated performance. By example LauncherOne had a stated payload performance of 500kg, but in its entire flight history it never lifted more than a few cube sats at a time.

3:44 PM · Aug 12, 2024 · 14.5K Views

8 14 237 16

## Unity Development

<https://github.com/CKalitin/Unity-Simple-Networking-Layer>

<https://github.com/CKalitin/Simple-Hex-Strategy-Game>

My final project in my Digipen Game Development class was an online multiplayer platformer called Stuck in the Simulation. After finishing this project I was familiar enough with C# Unity Networking programming that I decided to create a high-level Unity Package for networking. This abstracted away many of the intricacies of networking so that the user had an easier experience creating a game. At the end it was 8,000 lines of code and I implemented it in a game myself called Tiny Troops (second link).



*I'm no artist, excuse the extreme adherence to low poly art in this early build of the game.*

## PCB Design

For UBC Solar projects I taught myself PCB Design and designed a PCB based on the STM32 to record time series voltage and current data for easy characterization of various systems (eg. current sensor). Essentially an easily programmable multimeter & oscilloscope.

Currently doing this project, here's the current state of the PCB with firmware [here](#):

