

## Wright Scholar Essay (Topics 1 and 3)

294 squiggly red underlines. Nearly every line of my code had errors. Null pointers, incompatible types, undefined variables, Gradle sync errors—I had seen it all, and each error felt like a dissonant chord demanding resolution. It was February 2024, my freshman year, and we had ten minutes to take the field for our First Tech Challenge (FTC) final match. My heart raced as I thumped the keyboard in a frenzy. My code was broken, and for the final match, it had to work. Time raced faster than it ever used to, and I finally compiled the code. There was no time to test, hardly any to breathe, and before I knew it, we were on the field with my index finger hovering over the large play button. Time paused. I heard the buzzer and pressed play. Success. In two minutes and thirty seconds, we became league champions.

It was almost hard to believe that seven months back, I didn't know what a variable was. I was fully into music, and programming was not even an afterthought. It was a mere coincidence that my neighbor (and good friend) decided to start a robotics team, and given the minimal investment, I joined. Like nearly all of my endeavors, my FTC journey began with a Google search. I was learning at a snail's pace, and it had taken me two months to simply make a motor move. Soon, I was hooked. Like a sponge, I was absorbing everything I had to learn, and I had eventually taught myself enough Java to become a functional FTC programmer.

As the season progressed, my sponge was soaked, and we were a top competitive team by February. On competition day, the lightbulb within me finally clicked. The joy I experienced wasn't just from our robot picking up and scoring pixels, but from the fact that the code I had recently learned to write was resulting in a tangible output that I could witness. It was that moment when I decided to pursue a STEM career. I was no longer just a high school student; I was a STEM student, and I was ready to help change the world.

But that readiness was tested in September 2024. In a spur of ambitious insanity, I had committed to building a machine learning model to predict gait patterns in Parkinson's Disease for my sophomore-year Science Fair project. The problem: I had no clue how to. And so I learned. Python syntax, NumPy arrays, signal filtering, feature extraction, and model architectures. I had entered a brand new domain, and each concept seemed to confuse me in a different way. After two months of painfully laborious learning, coding, and debugging, I was finally able to transform raw sensor data into a functional and accurate classification model. Somewhere between the first error message and the final 96% accuracy, I had managed to absorb a new discipline by pushing myself into unfamiliar waters.

If it weren't for my ambition, I would have stopped there. Unfortunately, I realized that a working model on my laptop wasn't going to help any Parkinson's patients, and I needed to embed my model into a complete hardware device. This task was beyond daunting, as I had to venture into the foreign land of hardware and electrical engineering. With my engineering teacher guiding me, I slowly learned everything I needed. After dozens of 2 AM KiCAD tutorial binge sessions, I finally had a working design for a fully custom printed circuit board (PCB). Two weeks later, my PCB arrived, and after soldering all my components, it didn't work. My heart sank. I touched up all the joints with my soldering iron and tried again. Success. I wrote some quick software in C++, and I finally had a working end-to-end implementation for my final solution. After my project made it to the International Science and Engineering Fair (ISEF), the judges were impressed by the full end-to-end implementation, and my efforts were finally rewarded when I won 3rd in Robotics and Intelligent Machines at ISEF.

Throughout my high-school life, I have strived to constantly learn new things, which is why I am so excited about the Wright Scholar opportunity. From tinkering in my bed, I can start working on relevant research problems this Summer, which would exponentially increase my learning and comprehension of these subjects. From cutting-edge biomedical computing to sensor processing to cybersecurity, AFRL offers exciting venues for me to apply my knowledge. For a sponge who lives to learn, AFRL can serve as a reservoir of knowledge, and I can't wait to absorb new information with domain experts.