

# Isabelle André | Electrical Engineering Student

VANCOUVER, BC | [isabelle.andre14@outlook.com](mailto:isabelle.andre14@outlook.com) | (519) 897 0853

 [www.iandre.ca](http://www.iandre.ca)  [www.github.com/Abeilles14](https://www.github.com/Abeilles14)  [linkedin.com/in/isabelle-andre-ia](https://linkedin.com/in/isabelle-andre-ia)

## WORK EXPERIENCE

### Silicon Engineer — Microsoft

INTERNSHIP - MAY 2022 - AUG 2022, RALEIGH, NC

- Developed a transaction-based debugging interface using React JS to visualize L2 cache transactions, events, and attributes, facilitating design verification and tag entry debugging
- Correlated multithreaded cache ops on hardware level parsed to JSON
- Created stimulus to verify cache cleaning blocks by driving signals in UVM and adding support in the testbench for new RTL

### Automation Engineer — Apple

INTERNSHIP - JAN 2022 - APR 2022, CUPERTINO, CA

- Developed an automation framework to process output data from a variety of testers to a database, saving engineers hours of work
- Created new automation tools to reduce manual steps and errors and streamline compliance results across several product lines

### Software Integration Engineer — Tesla

INTERNSHIP - SEP 2021 - DEC 2021, PALO ALTO, CA

- Added coverage to the air suspension system SIL test infrastructure
- Developed unit tests for air suspension firmware and validated vehicle platform firmware releases in SIL infrastructure
- Lead the bring up and firmware integration of the Semi HIL tester

### Research Assistant — UBC Robotics & Controls Lab

DEC 2020 - DEC 2021, VANCOUVER, BC

- Automated robotic arms using task-based hierarchical state machines
- Implemented a decentralized motion planning framework for the collision avoidance of arms in a dynamic surgical environment
- Developed a task planning software for trilateral tasks using da Vinci Research Kit robotic surgery platform, ROS, and V-REP simulator

## ENGINEERING DESIGN TEAMS

### UBC Unmanned Aircraft Systems — Electrical Lead

- Developed the firmware and communication layer between Odroid, PWM, and brushed drivers to control the motor system of the aircraft
- Integrated remote PID control capabilities for drive, claw, and winch
- Developed a real-time QR scanning Python program using OpenCV to display location data on screen using an FPV camera system

### UBC Launchpad — Software Engineer

- Developed an iOS mobile application to track and classify a group's expenses with notifications using Swift, Node.js, and MySQL
- Built application infrastructure and backend API, and containerized the backend with a MySQL database using Docker

### UBC Rapid CAD and 3D Printing — Co-Captain

- Managed a team of 25+ students contributing to our Consulting, 3D printing, and 3D printer construction and maintenance business
- Lead the design and construction of a new compact & portable 3D printer

## EDUCATION

### University of British Columbia

SEPTEMBER 2018 - MAY 2023

4th Year Electrical Engineering, BAsC  
Vancouver, BC, Canada

### Language Skills

Verbal/Written: French, English

## TECHNICAL SKILLS

### Languages

Python, C++, C, Rust, Verilog, Assembly, Javascript, Matlab, Lua, Bash, Swift

### Frameworks

Node.js, React, Angular, Loopback, Ionic

### Other

Git, ROS, CAN, I2C, UVM, V-REP, SQL, Simulink, Solidworks, Docker, FPGA, Quartus, Modelsim, Linux

## PROJECTS

### Motion Planning and Velocity Collision Avoidance Framework

PYTHON, ROS, V-REP

Developed a task and motion planning framework to compute collision avoiding velocities such that each robot may avoid collisions without changing their initial trajectory

### Metal Collecting Autonomous Robot

C, MAKEFILES, EFM8, STM32F051, NRF24 TRANSCEIVER, BO230XS USB SERIAL

Designed and programmed a metal detecting robot collecting metallic objects automatically, restricted within a 0.5 m<sup>2</sup> perimeter wire, or manually using a controller

### FPGA MP3 Player

VERILOG, ASSEMBLY, MODELISM, QUARTUS, DE1 SOC

Designed an Music Player in System Verilog using FSMs to synchronize clks and read sound samples from flash memory, then writing to a D/A converter while interfacing with a keyboard