# Owen Brake

Waterloo Mechatronics Engineering - 3A · **OwenBrake.com** · (647)548-5493 · obrake@uwaterloo.ca

## Work Experience

### Parallel Systems
**Summer 2022**
*Electrical Engineering*
Los Angeles, CA

- Currently in Progress

### Tesla
**Fall 2021**
*Firmware - Drive Inverter Systems*
Palo Alto, CA

- Developed highly-performant, resource constrained firmware for the Drive Inverter boards
- Developed and deployed mission-critical features for millions of production vehicles
- Developed firmware across multiple chip architectures to accommodate for 2020/2021 Semiconductor Shortage

### Apple
**Winter 2021**
*Embedded Firmware Engineer*
Remote

- Specific features are currently redacted to preserve confidentiality

### Ford Motor Company
**Summer 2020**
*Software Engineer*
Remote

- Worked on system to process vehicle core dump files into easily readable, accessible and shareable online formats using GDB and Java
- Rewrote permissions system to enable complex and nested conditions while maintaining performance on system with over 1 billion database records in Java and SQL

### Groupdesk
**Summer 2019**
*Full Stack Developer*
Toronto, ON

- Developed CRUD services, using Angular to remove user dependence on technicians
- Automated front end QA using Go, Docker and Chromedp to increase release efficiency and stability

### Liberty Metrics
**Fall 2016**
*Data Entry*
Mississauga, ON

- Online data mining and compiling of hotel booking data

## Projects and Teams

### Waterloo Formula Electric Team (Head of Firmware)
**September 2019 - Present**

- Designed and implemented firmware for ARM Cortex-M7 and M0 boards in FreeRTOS and C which communicate on the CAN bus
- Developed sensor analytics platform on Python for Beaglebone to measure and visualize live vehicle performance remotely
- Worked on drivers for the various sensors and external boards on the car like: LTC6812, LTC6811, LTC4110, etc.

### Isidore, Custom Programming Language
**December 2019 - July 2020**

- Deployed JIT compiled, cross platform programming language built in LLVM using C++.
- Designed language to solve many of the runtime safety problems of C while retaining minimum overhead and lightning fast runtime performance

### Self Driving Go Kart
**June 2019 - August 2019**

- Utilized Arduino, motor controllers and RC radio to allow remote control of Go Kart.
- Produced computer vision and control software in OpenCV and Python

## Skills

| | |
|---|---|
| **Electrical:** | Soldering, Circuit Design, Op-Amp Filter Design, DMA, $I^2C$, SPI, UART, CAN, ARM Cortex-M |
| **Software:** | Altium, KiCAD, FreeRTOS, LLVM, Git, STM32CubeMX, PID, GDB/LLDB, MATLAB, SQL |
| **Programming:** | C, C++, Verilog, VHDL, ARM Assembly, Go, Python, Java, MERN/LAMP Stack |