Yaseen Mohamed

BASc. Mechatronics Engineering | University of Waterloo

y24moham@uwaterloo.ca | +1 (647)-833-3952 | github.com/y24moham | YaseenMohamed.netlify.app | Yaseen Mohamed

Skills

- **Software:** C/C++, Python (PEP 8, Pytest, Pandas), Data Structures & Algorithms, MATLAB, SQL, Git, Robot Framework.
- **Embedded/Firmware:** Embedded C; ARM Cortex-M, RTOS, DMA, timers/interrupts; Sensors; Linux (bash/CLI).
- **Platforms:** STM32 (HAL, LL), Arduino, ESP32, Raspberry Pi; FPGA (VHDL), PLC (Ladder Logic).
- **Interfaces/Debug:** I2C, SPI, UART, Wi-Fi, Bluetooth; JTAG/SWD, ST-Link, OpenOCD; DMM, Oscilloscope, Bench Power Supply.
- **Hardware/CAD:** PCB (Altium), Soldering; SolidWorks/Inventor/AutoCAD, GD&T, SimulationX; 3D printing (FDM/SLA/SLS).

Experience

Embedded Software Engineering Intern — Sept - Dec 2025

Aversan Inc.

- Debugged and extended **STM32H7 embedded C firmware** for a safety-critical aerospace device, following **DO-178** processes, committing verified changes through the team's **version control** workflow.
- Built **5 Python GUI** tools (PEP 8) over **UART** to parse and log test data, and automate firmware debugging/verification, uncovering tens of bugs and cutting test time by **70%**.
- Built a **Linux** Python GUI for **JTAG boundary-scan** (**OpenOCD**) to validate **solder/shorts** and connectivity, and perform **bit-banged I2C** to test peripherals **without needing MCU firmware**, catching manufacturing faults early and reducing rework.
- Implemented sensor/peripheral interfacing via **GPIO bit-banged I2C** on an STM32H7 using **DMA** and **HAL + custom timers**, plus **interrupt-driven UART RX** for status reporting, improving diagnostics and reducing component test time by **90%**.
- Created automated I/O tests in **Robot Framework** (via **RIDE**) on **Linux**-based **Automated Test Equipment (ATE)**, to verify set/read behavior and generate reports to be used for debugging, **FAA certification**, production, and maintenance.

Engineering Research Assistant — Jan - Apr 2025

University of Waterloo School of Optometry and Vision Science

- Built an **Arduino** + **48 ch MUX** + **custom PCB** + **Raspberry Pi** pipeline for a pressure-based footstep-detecting device.
- Redesigned the acquisition + export pipeline: built **C++** sensor data-collection code with decay compensation for the pressure-resistive material, and built a **Raspberry Pi Python CSV export pipeline**, boosting sampling frequency by **99%**.
- Built a **Python GUI** with a **grouping algorithm** to **preprocess** and visualize foot-pressure data, producing analysis-ready outputs.
- Designed/fabricated research equipment upgrades using **SolidWorks** and **FDM 3D printing**, cutting setup time by **80%**.
- Wrote detailed **technical documentation** and a **user guide** to support reproducibility and future equipment development.

Motion Capture Undergrad Research Assistant — Sept - Dec 2024

University of Waterloo RoboHub

- Calibrated the **Vicon** camera array and attached reflective markers per the Plug-in Gait protocol for precise spatial tracking.
- Monitored live capture at **200 Hz**, tuning exposure and thresholds to reduce occlusions and noise.
- Processed trials in **Vicon Nexus**: labeling marker trajectories, applying spline-based **gap filling** and **low-pass filtering**, and exporting cleaned files for biomechanical analysis.

Projects

EV3 LEGO Prosthetic Hand | GitHub · Demo | RobotC, SolidWorks, 3D Printing — Oct - Dec 2023

- Built a functional prosthetic hand in **RobotC (C-based)** using **EV3 motors/sensors** and a 3D-printed handle for controlled object handling as a hand replacement.

Quadcopter Prototype | GitHub · Demo | C++, Arduino, PCB/wiring, Controls (PID) — June - Aug 2020

- Built an Arduino quadcopter prototype with **PID stabilization** in **C++**, using **IMU (gyro) feedback** and custom **PCB/wiring** for real-time attitude control and tuning.

Obstacle-Aware Arduino Robot | GitHub · Demo | C++, Arduino, Real-time control — Oct - Dec 2023

- Built an obstacle-aware robot using **Arduino C++**, with an **LCD UI**, **ultrasonic ranging**, and button-controlled 2-state **finite-state machine** (idle/run) with debounce, in a simple real-time control loop for obstacle avoidance.

Education

University of Waterloo, BASc. Mechatronics Engineering — GPA: **3.9** | **Excellent standing** — Grad 2028

- **Relevant Coursework:** Embedded Systems/Microprocessors (C), OS & RTOS, DSA (C++/C), FPGA (VHDL), MATLAB, Circuits, Sensors and Actuators, OOP, Statistics, Statics/Dynamics.