Ryan McGee

Robotics / Automation Engineer

ram6064@rit.edu | 540.903.3552 | in ryan-mcgee-dev | ryanmcgee.dev

Searching for a full-time robotics / automation engineering position, starting May 2024.
In person / remote, willing to relocate if necessary .

Education

Rochester Institute of Technology (Senior) 2018-2024
Robotics and Manufacturing Engineering Technology | GPA 3.24

Employment

CACI - *Software Engineer Intern* - Dahlgren VA May 2019 - Dec 2021
- Developed C++ programs for capturing, processing and retransmitting network data
- Created JavaFX UIs to display live data inside a PostgresSQL database
- Generated packages and created regex bash scripts for RedHat systems

ITW Hartness - *Controls Engineer Intern* - Greenville, SC May 2022 - Dec 2022
- Wrote PLC ladder logic and created HMI UIs on Allen Bradley systems for accumulation machines
- Developed algorithms to control accumulation of product on mass-flow conveyors using vision sensors
- Used AutoCAD Electrical to develop panel schematics for robotic palletizing systems
- Simulated line product flow with the Emulate3D physics engine

ei3 Corporation - *Automation Engineer Intern* - Pearl River, NY June 2023 - Current
- Created a test stand to demonstrate platform-agnostic data ingestion on ei3 hardware
- Provisioned OPC UA servers and connections
- Implemented the PackML standard on Allen Bradley, Beckhoff and B&R platforms
- Created an HMI for controlling three PLCs at once using Maple Systems, following the PackML HMI standard

Skills

Automation:
- Ladder Logic
- Structured Text
- B&R Automation Basic
- HMI Design
- ABB Robot Studio
- ABB SMART Certification

CAD:
- Solidworks
- Autodesk Inventor
- KiCAD

Programming:
- C++
- Java
- Bash
- Git
- Linux

Frameworks:
- OpenCV
- JavaFX
- ESP-IDF
- ROS

Projects

3D Printed Robots: *ryanmcgee.dev/portfolio*
- Custom differential, swerve and mecanum drivetrains, designed in Inventor and FreeCAD
- Controlled via Raspberry Pi, Pi Pico and esp32 microcontrollers
- Programmed in Java, C and C++ (ROS), communicating over WiFi and Bluetooth

VEX Robotics CoreAPI *github.com/RIT-VEX-U/Core*
- A custom open source C++ API for easily managing mobile robotic subsystems and utilities
- Supports feedback loops, logging, and state-machine control with a modular, object-oriented approach
- Abstracts drivetrain and manipulator subsystems for a fast-paced development environment

Activities

RIT VEXU Robotics Team, *Treasurer and Software Lead*
- A robotics team that creates two mobile robots for a game released each year, and competes against other universities
- Developing software for motion controls and asynchronous state management
- Managing the GitHub organization's 5+ active repositories with code reviews, pull requests and wiki maintenance