Johnathan J. Flaggs

LinkedIn: www.linkedin.com/in/johnathanflaggs
Email: work.johnathanflaggs@gmail.com
Website: resume.johnathanflaggs.com
Phone: +1.949.414.9545

Robotics Software Engineer & Architect with a focus on backend system development for robotics applications. Experienced within numerous industries, languages, and technologies. Strong aptitude for mechanical design and has a successful record of leading projects from proof of concept into production.

Education

University of California, Riverside & Davis **Sep. 2010-Jun. 2014**

- UCR: BSME with concentration in Control Theory under Department of Mechanical Engineering (BCOE)
- UCD: Control Theory Concentration under Department of Mechanical and Aerospace Engineering

Technical Toolset

Development Languages & Environments:

▪ Visual Studio	▪ Codeblocks	▪ SVN, GIT, Jira	▪ Ignition SCADA	▪ Node.js
▪ Atmel Studio	▪ Embedded C, C++	▪ C# .NET	▪ MATLAB, Python	▪ HTML, CSS, JS
▪ RSLogix/FactoryTalk	▪ Beckhoff/TwinCAT	▪ Festo, MagneMotion	▪ Fanuc TP, Karel	▪ Cognex, iRVision
▪ RS-232/485, CANbus	▪ Allen Bradley	▪ IEC 61131-3	▪ gRPC, JSON, SQL	▪ EtherCAT, EtherNet/IP

Professional Experience

Lead Robotics Software Engineer at CarbonCapture **Jun. 2023-Present**

Driving system software architecture and automation development for modular Direct Air Capture (DAC) reactors. Ensuring that our team builds a software core that is sufficiently robust, scalable, and maintainable to support current and future product deployments. Leading hiring efforts, defining technical infrastructure, and building company culture to support a quickly growing technical team.

- **Identify** development resource needs and coordinate efforts with HR and VP to create new technical positions
- **Manage** sprint cycles and review source code on a per-commit basis to avoid common development pitfalls
- **Hosted** a bi-weekly coding program wherein Jr. and Sr. developers can learn and share their contributions
- **Establish** clear and achievable performance expectations and conduct regular performance evaluations
- **Synchronize** a team of mechanical, software, process, and data science engineers on a weekly basis
- **Interface** with contractors for outsourcing code and supplying measurement/control devices
- **Mentor** engineers on approaching complex problems and upkeeping best coding standards
- **Drive** design of macro and micro software architectures which define the core product
- **Anticipate** and mitigate the impact of future design changes on the software layers
- **Enforced** OOP and heavy emphasis on robustness, scalability, maintainability, and patterns in C#/C++, and TwinCAT
    - Increased software scalability/maintainability by refactoring code from FP to OOP
- **Drive** strategic decisions regarding budget, mechanical design, safety, maintenance, and scalability
    - Reduced materials budget by over $100k by eliminating need for absolute encoders
    - Eliminated expensive door motors by proposing pneumatic actuation design

Professional Experience

Lead Controls Engineer at Essentium 3D **Jun. 2021-Jun. 2023**

Leading robotics software architecture and development for high-speed industrial 3D printers. Ensuring that our team builds a software core that is sufficiently robust, scalable, and maintainable to support the product line. I refactored the existing code base into OOP scalable modules/layers. This allowed us to quickly change hardware and ultimately extend from plastics into metal printing.

- **Lead** the design and architecture of our core software which allowed us to expand from plastic to metal printing
- **Primarily** used Clearcore Teknic controllers for real-time control and C# for the application and UI layers
- **Mentor** Jr. Engineers on approaching complex problems and upkeeping best coding standards
- **OOP** and heavy emphasis on robustness, scalability, maintainability, and patterns in C# and C++
- **Anticipate** and mitigate the impact of future design changes on the software layers
- **Design** of macro and micro software architectures which define the core product
- **Manage** and review source control on a per-commit basis

Sr. Robotics Software Engineer at Seagate Technology **Mar. 2017-Jun. 2021**

Developing software for cutting edge processes in digital storage technology. I focused on developing complex motion control routines for pico-scale positioning, calibration routines, state-machine process synchronization, and machine vision integration using the Congnex SDK.

- **Primarily** used Beckhoff and Delta Tau for real-time control and C#/C++ for vision, application, and UI layers
- **Motion Control** Kinematics, pick-n-place, multi-axis coordination, Quantum HSM framework
- **OOP** Heavy emphasis on encapsulation, inheritance, polymorphism, and robust design patterns
- **User Interface** Allowing users to fluidly interact with the multi-threaded application
- **User Interface** Allowing users to fluidly interact with the multi-threaded application
- **Machine-Vision Calibration** to establish precision robot coordinate system
- **Version Control** Using SVN, TFS and Agile/Scrum using Jira
- **Support** core vision libraries in C/C++
- **Cognex** VisionPro API integration

Lead Controls Engineer at Sorenson Engineering Inc. **Sep. 2014-Mar. 2017**

Leading controls software development for high-volume manufacturing. I developed core software and worked closely with Mechanical Engineers to build proof of concept products (R&D environment).

- **Primarily** used Beckhoff, Festo, Maxxon, and Jenny Science for real-time control with TwinCAT for HMI/UI
- **Motion Control** – Kinematics, motor sizing, pick-n-place, multi-axis synchronization, and axis coupling
- **Increased** machine PPM by decreasing rotor inertia ratio by 140% for tighter position control (VSIII)
- **Fieldbus Integration** – Integrating multiple OEM device nodes into the EtherCAT controls network
- **Vision Inspection** – Driving digital cameras (Cognex, Baumer/VeriSens) via native C++ SDKs
- **Eliminated** need for expensive camming software license and a measurement sensor