Arpan Swaroop

402-913-5141 | aswar3@illinois.edu | linkedin.com/in/arpan-swaroop | github.com/aswaroop976

EDUCATION

University of Illinois at Urbana Champaign — Urbana Champaign, IL
B.S. Computer Engineering, Grainger College of Engineering — *Aug. 2021 – May 2025*
Relevant Coursework: Computer Systems Engineering, Database systems, Intro to Artificial Intelligence,
Multimedia Signal Processing, Digital Signal Processing, Data structures and Algorithms

TECHNICAL SKILLS

Languages: Python, C/C++, SQL (MySQL), JavaScript, HTML/CSS, System Verilog, x86 asm, Golang, Rust
Technologies: Node.js, Express, Google Cloud Platform, ROS, Nav 2, pandas, Matplotlib, FPGA
Developer Tools: Git, Docker, VS Code, Linux(Ubuntu)

EXPERIENCE

Research Assistant — Feb 2023 – Aug 2023
Advanced Controls laboratory
- Implemented path planning algorithms in quad-copters, to allow for autonomous flight
- Developed these algorithms using Euclidean signed distance fields for fast and flexible local planning
- Implemented the rapidly exploring random trees algorithm for efficient path planning
- Utilized **ROS** to integrate the depth sense cameras used for localization and planning
- Developed on the Nvidia Xavier nx platform

Illini Robotics in Space autonomous team — Aug 2022 - May 2023
University of Illinois at Urbana Champaign — *Champaign, IL*
- Specialised in writing code which allowed a lunar rover to navigate a lunar environment autonomously In order to participate in the Lunabotics competition hosted by NASA
- Contributed to a physics simulation model of the robot in Gazebo
- Configured odometry to aid in the simultaneous localization and mapping of the robot
- Configured a cost-map using the **Nav 2** library in the **ROS** ecosystem
- Configured **Nav 2** plugins, and a behavior tree, to utilize navigation algorithms and complete navigation tasks

PROJECTS

Unix Like 32-bit x86 Operating System | *C, x86 asm*
- Architected a minimalistic multitasking operating system for x86 hardware, prioritizing efficiency and reliability
- Devised an interrupt-driven, time-slice-based CPU scheduling algorithm to schedule process execution
- Supported virtual memory using paging data-structures
- Developed a custom filesystem that supports basic file operations, utilizing storage, and I/O principles
- Implemented device drivers for keyboard, Real Time Clock, Programmable Interrupt Controller, and Programmable Interval Timer

Game Developer's toolkit | *Javascript, HTML, CSS, Node.js, Express, MySQL, Python*
- Developed full-stack web application using a steam game database to help game developers
- Implemented multiple advanced queries to the database to output useful information for game developers
- Implemented interactive graphs in **Python**, showing helpful patterns in game data across all steam games
- Added features to enable user sign in and login, and the ability to add other users as friends
- Utilized a **REST API** made with **Node.js** and **Express**
- Hosted website on the Google Cloud Platform

Chip-8 Emulator and Microcontroller port | *Rust, C*
- Developed a Chip-8 emulator in Rust, Engineered to be resource-efficient, optimizing for constrained environments
- Ported the emulator to an ARM Cortex-M4 microcontroller, adapting Rust code for embedded systems
- Implemented I/O functionality on the microcontroller, managing keyboard input and display output via **SPI** interface
- Ensured real-time performance and responsiveness by utilizing hardware timers and **interrupt-driven** design

Chat-App | *Golang, javascript, HTML, CSS*
- Created a simple chat-app utilizing a backend http web-server written in Go
- Utilized **web-sockets** with **Go-routines** and **Go-channels** to allow multiple users to connect to the server
- Utilized javascript, HTML, CSS to create dynamic webpages served from the web server