Tom Schammo

 Tom Schammo | Tom Schammo | tom@tomschammo.com | +352 621 48 0040

EDUCATION

2019 - 2024   B.Sc. (Computer Science) at **University of Tübingen**   (Grade: 2.36)

Courses include:

– Programming Ultra Low Power Architectures: A M.Sc. course, introducing students to low power states, hardware drivers, buses and interrupts.

– Massively Parallel Computing: A M.Sc. course providing students with the skills to utilize GPUs and CUDA to more efficiently solve various problems including matrix multiplication, manipulation of data structures on the GPU and NBody systems.

– Teamproject: A course teaching students how to plan and work on a project as well as collaborating using version control (git).

– Thesis: For my thesis I aimed to enhance (audio) keyword spotting on embedded devices.

WORK EXPERIENCE

University of Tübingen   Nov 2023 - Feb 2025
Research Assistant

As part of a self-driving car research project, I took full ownership of the development of a Lidar point cloud augmentation library. The library gathers different SOTA augmentation methods and utility functions in an easy-to-use Python frontend whilst significantly improving the performance due to the efficient C++ backend.
As the sole contributor, I was responsible for conceptual planning, programming, testing and maintenance of the library, developing my skills to independently conduct medium-scale projects end-to-end.

Octoshrew Ltd.   Nov 2022 - Nov 2023
Junior Developer

As a junior developer, I took part in developing state-of-the-art path planning and autonomous control systems for drones. Specifically being responsible for pathfinding, I developed the data structures and algorithms to assure safe and efficient navigation in 3D space, thereby improving my proficiency in algorithms and performance optimization/parallelization.

PROJECTS

Implementation and Analysis of different Input Sources on the Ultratrail Architecture

For my thesis I worked on implementing a Rust driver for the Ultratrail AI accelerator on the PULPissimo board (a development board for a RISC-V chip designed by the ETH Zürich) to enable efficient keyword spotting in audio.
By innovating upon the existing C driver by improving security using the Rust programming language, I improved my proficiency and extended my knowledge in embedded programming as well as debugging on a hardware level using tools such as oscilloscopes or logic analyzers.

Programming Ultra Low Power Architectures

I built an alarm clock proof-of-concept using a development board and peripherals such as a small LED display, a keypad and a speaker. The clock took advantage of deep sleep states to be very power efficient, surviving 12 hours on a small capacitor alone. As a reference, on one AA battery the clock would last 130'000 to 220'000 hours in power-saving mode.
The project provided an opportunity to gain insight into the use of hardware documentation as well as the development and debugging of hardware drivers and embedded applications. Additionally, it offered a chance to explore the use of processor sleep states and tools for measuring power consumption resulting in improved battery-powered designs.

SKILLS

Languages:	English (C2), German (native), Luxembourgish (native), French (B2)
Programming Languages:	C, C++, CUDA, Python, Rust, RISC-V assembly, Mips assembly
Developer Tools:	Make, CMake, Git, GitHub, GitLab, Linux, Docker, Logic Analyzer, Oscilloscope
Frameworks:	Libtorch, Boost, Google Test, ROS, PyTest, Numpy