

Ulaş Tan Ersoyak

Mol, Belgium • (+32) 470 48 34 65 • ulassztanersoyak@gmail.com • github.com/ulastanersoyak • ulastanersoyak.com

EDUCATION

MSc Computer Science — University of Antwerp

Sep 2026 – expected Jun 2028, Antwerp, BE

BSc Software Engineering — Toros University

Feb 2021 – Jun 2025, Mersin, TR

GPA 3.45 / 4.00 — Thesis: Design and Implementation of a Linux Process Manipulation and Runtime Analysis Tool

Erasmus+ Exchange (Computer Science) — AGH University of Krakow

Winter semester 2022, Krakow, PL

EXPERIENCE

Intern — TÜBİTAK BİLGEM İLTAREN

Jan 2025 – Jun 2025, Ankara, TR

- Developed bare-metal drivers for avionics systems of the KAAN fighter jet, including a power hold-up module and a heat sensor.
- Built IP cores for encryption and decryption using Vitis HLS (FPGA high-level synthesis).
- Created custom embedded Linux kernels and images with PetaLinux and Yocto.

Intern — Gürbağ Savunma

Jul 2023 – Aug 2023, Ankara, TR

- Developed a position estimation system on a custom ARM microprocessor.
- Integrated GPS and IMU sensor modules for accurate tracking and motion analysis.
- Implemented sensor fusion algorithms to combine GPS/IMU data and estimate target position.

Intern — Mersin Teknopark

Feb 2022 – Sep 2022, Mersin, TR

- Tested autonomous driving functionalities of the company's robots in a ROS 2 simulation environment.
- Developed motor drivers for a robot prototype.
- Built Git hooks for code formatting and static code analysis in the company's CI/CD pipeline.

PROJECTS

moOS — 32-bit Operating System Kernel

github.com/ulastanersoyak/moOS

- Implemented a kernel from scratch: custom bootloader, GDT/IDT setup, interrupt and exception handling, and a dynamic heap allocator.
- Wrote drivers for VGA text mode, keyboard and disk I/O; built a FAT16 filesystem parser/loader and an interactive shell.

pp — Linux Process Debugging & Runtime Analysis Tool

github.com/ulastanersoyak/pp

- Built a runtime analysis tool that reads/writes process memory, scans and patches byte patterns, modifies memory permissions and inspects threads.
- Implemented ELF symbol resolution and Capstone-based disassembly; added shared-library injection and experimental function hooking via injected machine code and syscall stubs.

jet_fighter — Atari 2600 Game in 6507 Assembly

github.com/ulastanersoyak/jet_fighter

- Built a playable Jet Fighter clone entirely in 8-bit 6507 assembly, working within the Atari 2600's 128 bytes of RAM; assembled with dasm and tested on the Stella emulator.

SKILLS

Languages & tools: C, C++, Python, CMake, Make, Git, Linux, Yocto/PetaLinux

Domains: Embedded systems, bare-metal development, kernel/OS internals, communication protocols (UART, I2C, SPI)

LANGUAGES

Turkish (native) • English (C1 — IELTS 8.0) • German (A1)