Address	Mail	GitHub	Website	LinkedIn
Istanbul, Turkiye	*contact@msakg.com*	*@imsakg*	*https://msakg.com*	*www.linkedin.com/in/msakg*

Mert Sefa AKGUN

Software Developer

Summary

Hi, I'm Mert, a passionate Software Developer and Open-Source enthusiast with a deep-seated drive to learn, explore, and tackle new challenges. With a strong background in writing production-quality firmware and software, I have cultivated a particular interest in embedded systems, real-time systems, systems programming, robotics, and autonomy with keen eye of interest to fields such as aviation, space, defense industries, and electronics, where I am eager to contribute and make an impact. As an independent learner and self-motivated tech enthusiast, I have developed a solid foundation in low-level programming and embedded systems, always striving to push the boundaries of what technology can achieve. I am confident that my skills and passion can bring significant value to your organization, and I look forward to the opportunity to collaborate and drive innovation together, taking the company to new heights.

Work Experience

Embedded System Software Developer (Internship)
March, 2024 - June, 2024

Baykar

Created Real-time Networking Infrastructure for Military-level Aerial Vehicles

- Utilized the IEEE 802.1 TSN (Time-Sensitive Networking) specification as a guide to develop real-time networking infrastructure.
- Implemented Frame Replication and Elimination for Reliability (IEEE 802.1CB) on OSI Layer 2 with minimal overhead by leveraging cutting-edge technologies such as eBPF (extended Berkeley Packet Filter) and XDP (eXpress Data Path).
- Worked with IEEE standards, including the generalized Precision Time Protocol (IEEE 802.1AS), Time-Aware Shaper (IEEE 802.1Qbv), and Frame Pre-emption (IEEE 802.1Qbu).
- Utilized Linux build tools (Yocto/Buildroot) to achieve real-time networking capabilities at the kernel level.

Stack Set: extended-Berkeley Packet Filter, eXpress Data Path, Linux, FreeRTOS, Bare-metal, Real-time Networking Infrastructure, OSI 2-3, Rust, C/C++, Yocto, BuildRoot.

Embedded System Software Developer (Full-time)
March, 2022 - March, 2024

Fora

Worked on all variants of Mavisoft (Access Control System) product line and other large building/enterprise security/safety products.

- Development of versatile, reliable and precise embedded access control softwares on baremetal/RTOSs for brand new products.
- Developed a C/C++ library for Open Supervised Device Protocol (OSDP).
- Developed a symmetric encryption/authentication library in C for devices that do not support the integration of off-the-shelf TLS libraries. The implementation was verified by a 3rd party security audit company.
- Implemented many feature requests from customers in Mavisoft Access Control System.
- Maintaining code, dealing bugs like UB, addressing critical issues reported by site/customers.
- Developed lots of peripherasl device for different use cases like MiFare, HID, Proximity, NFC card readers.

Developed RTLS (Real-time In-door Localization System) that alternative to GPS.

- Worked with (UWB) ultra-wide band signals
- Implemented TDoA, TWR positing algorithms over systems.
- Worked with BLE stack.
- Developed client-side application for visualizing and configuring.
- Implementation of filtering, estimator algorithms such as EKF (Extended Kalman Filter)Worked on all variants of Mavisoft (Access Control System) product line and other large building/enterprise security/safety products.

Stack Set: UART/USART, I2C/I2S, SPI, ModBus, RS232/RS485, TCP/IP Stack. MCUs (ST, Nuvoton, Renesas, ESP, Nordic, Atmel), Serial to Ethernet, Zephyr, Nuttx, FreeRTOS.

System Administrator
Jan, 2017 - Present

Self-employed

- *Hosted numerous low to medium-scale client websites and projects on dedicated servers at providers such as AWS, GCP, and Hetzner.*
- *Managed and optimized server resources to meet specific client requirements.*
- *Created secure client access by setting up network tunnels, SSH jump servers, and NGINX reverse proxies for web and shared server instances.*
- *Configured a two-node Proxmox cluster with RAID 10 and in-memory replication for enhanced reliability.*
- *Fine-tuned KVM/QEMU for improved efficiency and fair resource sharing.*
- *Configured Ceph for distributed block storage to ensure data redundancy and high availability.*
- *Utilized Opnsense for advanced firewall management, NAT/TC policing, IP tables configuration, and VLAN routing.*
- *Set up VPNs using WireGuard and OpenVPN-DCO to optimize efficiency.*
- *Web servers used such as NGINX and Apache, and databases including PostgreSQL, MySQL, and MongoDB.*
- *Configured a multi-protocol (IMAP/SMTP/POP) mail server with TLS for enhanced security.*

Academic History

Cumhuriyet University

BEng Degree
2020 - 2024

Modules included : Algorithm Analysis, Data Structures, Object Oriented Analysis, Numerical Analysis, Physics, Differential Equations, Electronic Circuits and Design, Computer Architecture, Numerical Analysis, Automata Theory, Signals & Systems, Operating Systems, Computer Networks, Embedded Systems, Microcontrollers, Database Systems, Cyber Security, Machine Learning, Probability and Statistics, Cloud Technologies.

Activities & Accomplishments

- *Google Summer of Code 2020 - Python Fury*
- *Teknofest 2021 Unmanned Under-water Vehicle Competition - Finalist (7th)*
- *Teknofest 2021 International Unmanned Air Vehicle Competition - Finalist*
- *Teknofest [2022,2023] International Fighter Unmanned Air Vehicle Competition*

Main Interest

I am passionate about conducting research in evolutionary algorithms, genetic programming, and autonomy through genetic algorithms. I am also deeply interested in distributed network protocols, RF systems, and cryptography. In essence, I enjoy working with any programmable systems. Additionally, I am a maker who enjoys designing circuit boards and creating 3D models.

Additional Skills

Social Skills

I consider myself a results-driven individual who is committed to ensuring that projects are successfully completed and fully functional. I have a tendency to over-engineer solutions because I believe that attention to detail is crucial. Additionally, I am adaptable and highly collaborative, making me an effective team player

Language Skills

English (IELTS 6.5), Turkish (Native), German (Very limited)

Technical Skills

Programming Languages : I'm using C&C++, Python, Rust in my daily life, also using Dart, Java, VB.NET, C# languages on such of projects when i need them.

Data management : Relational databases (PostgreSQL, MySQL), Key-Value Stores (Redis) and Document stores (MongoDB), Real-time Databases (Google's Firebase, Supabase)

Embedded development : Experienced with Espressif (ESP32, ESP8266), Atmel (ATmega328P, SAM3X8E, SAMD21), STM (STM32F1, STM32F4, STM32H7) Nuvoton (NUC029), Nordic (nRF52832) ,Renesas (RA4M1) micro-controllers and SBCs like Nvidia Jetson (Nano, Xavier), Raspberry PI 3-4-5, Texas Instruments

Technologies: Familiarity with some cloud infrastructure providers (AWS: EC2, S3, Lambda & GCP : Cloud Engine, App Engine, Cloud Storage). Limited experience with ML tools such as PyTorch, Tensorflow. Web Frameworks or APIs (Django, FastAPI, Actix, Rocket).

Tools : Linux (Mostly spending my time on lovely Arch, also we have long history with Debian), NeoVim, Emacs, Bash, Selenium, Visual Studio, VS Code, IAR&GCC Toolchains, OpenCV

Production: Some of Adobe products (Photoshop, Premiere, Lightroom, InDesign, Audition) Magix Vegas Pro, FL Studio

3D/Electronics: Autodesk Fusion 360, FreeCad, KeyShot, Ultimaker Cura, Blender, KiCad

Projects

Autonomy & Robotics

Flight Controller Software

Flight-controller software that running on bare-metal(STM32F407VET6) for drones.

- *Built on Mbed OS which is Real-time Operating System by ARM.*
- *Extended Kalman Filter is available for main state estimator.*
- *Implemented lot of variants IIR and FIR to sensors output as DSP algorithms.*
- *Kalman based PID controller model tuned to be ready to use.*
- *Implemented lot of hardware abstraction layer for couple of sensors.*
- *Using 192-bit ChaCha20-Poly1305 and S-Box based PRESENT Chipper for encrypted communication.*
- *NRF24L01 libraries implemented for RC communication.*
- *GCS (Ground Control Station) for visualizing data that coming over telemetry. It's capable to set parameter of vehicle.*
- *One pair transmitter and receiver for wireless communication.*

System Crafting

Oligarchy - x86 Kernel Project

Oligarchy is an X86 toy-kernel project. It's an operating system kernel I started to develop in C++ but later I decide to make it with Rust cause it's ownership model and fearless concurrency paradigms. Also it has more performance, guaranteed memory safety (barrow checker) and smaller footprint. At this point, it is not even close to feature-complete. I mostly use it as a testbed to try out interesting ideas. The kernel is developed for the x86 architecture. For now, no other architectures are planned due to the lack of manpower.

- *PCI support.*
- *Network Stack.*
- *Thread Scheduler.*
- *64-bit higher half kernel.*
- *4/5 level on-demand paging.*
- *Preemptive per-cpu scheduler.*
- *Modern UEFI bootloader.*
- *ACPI support (ioapic, lapic).*
- *Symmetric Multiprocessing (SMP).*