

KICHIK HAJMLI IOT QURILMALARDA MIKRO-YADROLI (MICROKERNEL) TIZIMLARNI RUSTDA YOZISH TAJRIBASI

*Farg'ona davlat universiteti amaliy matematika
va informatika kafedrasi o'qituvchisi*

Umarov Bekzod Azizovich

ubaumarov@mail.ru

FarDU Fizika-matematika fakulteti

23.11-guruh talabasi

Xamidov Hojiakbar

hojiakbarxamidov1@gmail.com

Annotatsiya: Ushbu maqolada resurslari cheklangan kichik hajmli IoT qurilmalar uchun mos bo'lgan mikro-yadroli (microkernel) operatsion tizimlarni ishlab chiqishning nazariy va amaliy jihatlarini ko'rib chiqiladi. Xususan, Rust dasturlash tilining xotira xavfsizligi, samaradorligi va modullikka asoslangan yondashuvi microkernel arxitekturasiga qanday uyg'unlashishi tahlil qilinadi. Tock OS va Redox OS kabi Rustda yozilgan tizimlar misolida tajriba asosida yondashuvlar o'rganiladi. Maqola IoT qurilmalar uchun ishonchli, xavfsiz va energiya tejankor operatsion tizimlar yaratishda yangi imkoniyatlarni ochib beradi.

Kalit so'zlar: IoT, microkernel, Rust, operatsion tizim, xotira xavfsizligi, Tock OS, Redox OS, tizimli dasturlash, embedded systems, real-time tizimlar

Annotation: This article examines the theoretical and practical aspects of the development of micro-core (microkernel) operating systems suitable for IoT devices with limited resources. In particular, it is analyzed how the memory security, efficiency, and modularity-based approach of the Rust programming language harmonizes with the microkernel architecture. Experimental approaches are explored using the example of Rust-written systems such as Tock OS and Redox OS. The article will open up new opportunities in creating reliable, safe and energy efficient operating systems for IoT devices.

Keywords: IoT, microkernel, Rust, operating system, memory security, Tock OS, Redox OS, systematic programming, embedded systems, real - time systems

Аннотация: В этой статье рассматриваются теоретические и практические аспекты разработки микроядерных (микрояdroвых) операционных систем, подходящих для IoT-устройств с ограниченными ресурсами. В частности, анализируется, как безопасность памяти, эффективность и модульный подход языка программирования Rust сочетаются с архитектурой микроядра. Экспериментальные подходы рассматриваются на примере таких написанных на Rust систем, как Tock OS и Redox OS. Статья откроет новые возможности для

создания надёжных, безопасных и энергоэффективных операционных систем для IoT-устройств.

Ключевые слова: IoT, микроядро, Rust, операционная система, безопасность памяти, Tock OS, Redox OS, систематическое программирование, встраиваемые системы, системы реального времени

Kirish

Bugungi kunda Internet of Things (IoT) qurilmalari hayotimizning ajralmas qismiga aylangan: aqlli soat, smart-sensor, uy avtomatikasi, sanoat monitoring tizimlari — bularning barchasi yuqori samaradorlik va xavfsizlikni talab qiladi. Bunday qurilmalar odatda resurslari cheklangan, energiyaga tejamkor, kichik hajmli mikrokontrollerlar asosida ishlaydi. Ana shunday muhitda ishlaydigan operatsion tizimlar uchun mikro-yadroli (microkernel) arxitektura va xavfsiz dasturlash tillari, xususan Rust, ayni muddao bo'lib xizmat qilmoqda.

Mazkur maqolada IoT qurilmalarda mikro-yadroli tizimlarni yaratish g'oyasi, Rust tilining bu boradagi afzalliklari, mavjud tajribalar va istiqbolli yondashuvlar haqida fikr yuritiladi.

IoT qurilmalarining cheklovlari va ehtiyojlari

IoT qurilmalar o'z tabiatiga ko'ra quyidagi cheklovlarga ega:

- Past hisoblash quvvati – ko'p hollarda ARM Cortex-M singari mikrokontrollerlardan foydalaniladi.
- Xotira cheklovi – RAM va ROM juda kichik hajmda bo'ladi (ko'pincha 64 KB dan kam).
- Energiya samaradorligi – qurilma uzoq vaqt davomida batareyada ishlashi kerak.
- Real vaqt rejimi – sensorlar yoki aktuatorlar bilan ishlashda kechikishlarga yo'l qo'yilmasligi lozim.

Shu sababli, bunday qurilmalar uchun engil, xavfsiz, barqaror va resurs tejamkor operatsion tizim zarur.

Microkernel arxitekturasi: IoT uchun ideal yechim

Microkernel – bu operatsion tizim yadrosining minimalistik shakli bo'lib, u faqat eng zarur komponentlarni (masalan, kontekst almashish, inter-process communication, signal handling) yadro darajasida bajaradi. Qolgan barcha xizmatlar (fayl tizimi, drayverlar, tarmoqlar) foydalanuvchi rejimida ishlaydi.

Microkernel afzalliklari:

- Modullilik – tizim komponentlarini alohida boshqarish mumkin.
- Barqarorlik – bir modul ishdan chiqsa, boshqalarga ta'sir qilmaydi.
- Xavfsizlik – xizmatlar ajratilgan rejimda bo'lgani uchun hujumlar yadroga yetib

bormaydi.

- IoT uchun qulaylik – minimal hajm, zaruriy komponentlar, energiya tejamkorlik.

Rust dasturlash tili: Tizimli darajadagi xavfsizlik

Rust — zamonaviy, past darajadagi imkoniyatlarga ega bo'lgan, xavfsizlikka yo'naltirilgan dasturlash tili. U tizimli dasturlashda kengayib borayotgan tildan biriga aylandi.

Rustning asosiy afzalliklari:

- Memory safety (xotira xavfsizligi) — Rust kompilyator darajasida pointer xatoliklari, buffer overflow va null-pointer'larni oldini oladi.

- No garbage collector – xotira avtomatik boshqarilsa-da, u real vaqt tizimlariga mos keladi.

- Concurrency safety – ko'p oqimli dasturlar xavfsiz boshqariladi.

- Embedded Systems uchun kutubxonalar – `no\_std` rejimda ishlash, `embedded-hal` interfeysi kabi imkoniyatlar mavjud.

Rust, C/C++ ga nisbatan xavfsizroq, ammo samaradorlikda raqobatdosh tildir.

Ayniqsa, tizimli dasturlashda xavfsizlik muhim bo'lsa, Rust eng yaxshi tanlov hisoblanadi.

Amaliy tajribalar: Tock OS va Redox OS misolida

Rust asosidagi bir nechta real tizimlar mavjud bo'lib, ular microkernel arxitekturasiga asoslangan:

Tock OS

- IoT qurilmalar uchun yaratilgan microkernel tizimi.

- Har bir drayver yoki servis alohida "capability" orqali ishlaydi.

- `no\_std`, `embedded-hal`, `Cortex-M` mikrokontrollerlari uchun moslashtirilgan.

Redox OS

- Rustda yozilgan to'liq operatsion tizim.

- Desktop darajasida ishlaydi, ammo arxitekturasida microkernel.

- Barqarorlik va xavfsizlikda yuqori ko'rsatkichlarga ega.

Bu tizimlar tajribasi shuni ko'rsatadiki, Rust tilida yozilgan microkernel tizimlar nafaqat mumkin, balki samarali yechimdir.

Xulosa

Kichik hajmli IoT qurilmalar uchun microkernel arxitekturasida – modul, barqaror va xavfsiz bo'lgan engil tizim yaratish imkonini beradi. Rust tili esa xotira xavfsizligi

va resurs tejamkorligi jihatdan bu ehtiyojlarga javob bera oladi. Hozirda Tock OS kabi real loyihalar bu yondashuvni amaliy jihatdan isbotlab bermoqda.

Kelajakda, ayniqsa sanoat IoT, tibbiyot qurilmalari, va real-time tizimlar uchun Rust asosidagi microkernel'lar asosiy yechimlardan biri bo'lishi mumkin. Mazkur yo'nalishda mustaqil tadqiqotlar, tajriba sinovlari va optimallashtirish usullari ustida ishlash ilmiy va amaliy ahamiyat kasb etadi.

Foydalanilgan adabiyotlar

1. <https://www.redox-os.org/> – Redox OS loyihasi (Rust asosida yozilgan operatsion tizim)
2. <https://www.tockos.org/> – Tock OS: IoT qurilmalar uchun Rustda yozilgan microkernel tizimi
3. The Embedded Rust Book – <https://docs.rust-embedded.org/book/>
4. Operating Systems: Three Easy Pieces – Remzi H. Arpaci-Dusseau, Andrea C. Arpaci-Dusseau
5. Rust Programming Language – <https://doc.rust-lang.org/>
6. Tanenbaum A. Modern Operating Systems. 4th Edition.