

Naslov teme

Analiza performansi i verifikacija real-time operativnog sistema na RISC-V arhitekturi

Kratki opis rada

Ovaj rad se bavi analizom performansi i verifikacijom real-time operativnog sistema (RTOS) na RISC-V arhitekturi, uz primjenu formalnih metoda i matematičkih modela za procjenu determinističkog ponašanja sistema. RTOS se široko koristi u ugradbenim sistemima koji zahtijevaju predvidljivo vrijeme izvršavanja, uključujući industrijsku automatizaciju, robotiku i sigurnosno kritične sisteme.

Rad će obuhvatiti sljedeće aspekte:

1. Analiza ključnih RTOS koncepata:

- Upravljanje zadacima: Istraživanje metoda za kreiranje, raspoređivanje i završavanje zadataka unutar RTOS-a. Analiza načina na koji RTOS upravlja višestrukim zadacima i osigurava da se svaki zadatak izvršava u predviđenom vremenu.
- Upravljanje memorijom: Pregled strategija koje RTOS koristi za efikasno upravljanje memorijom, uključujući alokaciju i dealokaciju memorijskih resursa.
- Sinhronizacija: Istraživanje mehanizama sinhronizacije, poput mutexa i semafora, koji se koriste za koordinaciju među zadacima.
- IPC mehanizmi (Inter-Process Communication): Analiza metoda komunikacije između različitih zadataka unutar RTOS-a.

2. Matematička analiza vremena izvršavanja i determinističkog ponašanja sistema, uz primjenu:

- Worst-Case Execution Time (WCET) analiza: Procjena najgoreg mogućeg vremena izvršavanja zadataka kako bi se osiguralo da svi zadaci završe u predviđenom vremenu. Ova analiza omogućava identifikaciju zadataka koji mogu uzrokovati kašnjenja i optimizaciju njihovog izvršavanja.
- Liu & Layland model: Primjena ovog modela za analizu determinističkog raspoređivanja zadataka. Model omogućava procjenu raspoloživosti sistema za izvršavanje zadataka u realnom vremenu.
- Teorija redova čekanja (Queuing theory): Modeliranje performansi međuprocenjske komunikacije kako bi se procijenila efikasnost IPC

mehanizama i identifikovale potencijalne mane, ili dijelovi sistema koji su najranjiviji.

3. Eksperimentalna evaluacija performansi kroz:

- Testiranje upravljanja timer-ima, prekidima i algoritmima raspoređivanja zadataka: Ispitivanje različitih algoritama raspoređivanja zadataka, kao što su Rate Monotonic Scheduling i Earliest Deadline First, kako bi se procijenila njihova efikasnost i uticaj na performanse sistema.
- Praktična implementacija i testiranje: Implementacija RTOS-a na RISC-V platformi koristeći FreeRTOS, te evaluacija performansi na stvarnom hardveru (Raspberry pi Pico board) ili emulatoru (najvjerojatnije QEMU emulator). Testirati kako RTOS upravlja task-ovima, timer-ima, prekidima i sinhronizacijom u različitim scenarijima.

Očekivani rezultati

- Detaljna analiza performansi FreeRTOS-a na RISC-V arhitekturi.
- Identifikacija i optimizacija ključnih aspekata upravljanja zadacima, memorijom, sinhronizacijom i međuprocenom komunikacijom.
- Praktični doprinosi kroz implementaciju i testiranje FreeRTOS-a na RISC-V platformi.
- Teoretski doprinosi kroz primjenu formalnih metoda za procjenu determinističkog ponašanja sistema.

Rad će imati teoretski i eksperimentalni doprinos, s ciljem istraživanja kako formalne metode mogu poboljšati verifikaciju real-time sistema i osigurati njihovo determinističko ponašanje u različitim scenarijima.

Preporučena literatura

1. Jean J. Labrosse, MicroC/OS-II: The Real-Time Kernel, CMP Books, 2002.
2. Giorgio C. Buttazzo, Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications, Springer, 2011.
3. Richard Barry, Mastering the FreeRTOS Real-Time Kernel, Real Time Engineers Ltd, 2021.
4. FreeRTOS zvanična dokumentacija: <https://www.freertos.org/>

Zvanični tutorijali:

1. [Driver Development for Zephyr RTOS and nRFConnect](#)
2. [Using FreeRTOS on RISC-V](#)
3. [QEMU Emulator Install and Start Guide](#)
4. [What is an RTOS? RTOS Tutorials](#)
5. [FreeRTOS Tutorials](#)