

Procesoare de timp real.

Driver de sistem și planificarea sarcinilor

Liviu Dumea

Rezumat

Aplicații de timp real nu sunt obligatoriu și aplicații de înaltă performanță. Mai importante decât timpul de execuție și memoria utilizată sunt determinismul și predictibilitatea sistemului. Este esențial ca în orice moment să fie cunoscută starea sa pentru a putea anticipa modul în care acesta va reacționa la evenimentele din exterior. Atunci când sunt proiectate astfel de sisteme se mai dorește să se evite întârzierile, să se obțină latențe cât mai scăzute și toleranță ridicată la defecte.

În general astfel de sisteme fac parte din aplicații complexe. Sunt necesare procesoare care să ruleze programe cu performanțe ridicate dar în același timp dorim și un control riguros, de timp real, al componentelor hardware. Cu alte cuvinte, este nevoie atât de procesoare de uz general cât și **procesoare de timp real**.

De cele mai multe ori aceste procesoare sunt separate. Pentru a se evita creșterea costurilor și a timpului de transmisie a datelor între acestea, producătorii au reunit pe aceeași pastilă de siliciu ambele tipuri de procesoare. Un exemplu de astfel de procesor este Sitara **AM335x** care pe lângă nucleele de uz general conține și nuclee de timp real, cunoscute cu denumirea de **unități programabile de timp real (PRU)**.

Proiectul își propune să creeze un driver de sistem pentru **procesoarele PRU**, ce poate fi folosit pentru dezvoltarea de aplicații ce rulează în cadrul sistemului de operare **Linux**. Spre deosebire de soluțiile existente se adaugă posibilitatea rulării mai multor fire de execuție. Pentru a obține acest lucru este nevoie de realizarea unui **planificator pentru sarcini** ce rulează pe unitățile de timp real. Aceste procesoare diferă prin arhitectură de procesoarele de uz general iar implementarea proiectului este influențată de acest lucru. De exemplu, lipsa întreruperilor face ca planificarea sarcinilor să fie cooperativă.

Un mod de a ne imagina tipul de aplicații specifice pentru aceste procesoare, este de a le privi ca **periferice** realizate software. Astfel se pot reduce costurile pentru proiectarea de noi dispozitive hardware. Pentru organizarea aplicațiilor este utilizată **convenția și nu configurarea**. Cele mai multe informații sunt extrase din codul sursă scris în **limbajul C**, respectiv din fișierele compilate în formatul **ELF**. Sistemul este capabil să analizeze aceste date și să ruleze aplicațiile fără a fi nevoie de a duplica informațiile pentru configurare.

Pentru a putea realiza un astfel de sistem sunt necesare cunoștințe în special legate de arhitectura sistemelor de calcul și a sistemelor de operare. Acumularea de informații despre structura unui procesor, planificarea sarcinilor și driverele pentru sistemul de operare Linux, prin realizarea unui sistem mai complex care să utilizeze aceste elemente, reprezintă un pas important în evoluția ca inginer în domeniul Calculatoarelor.