

PROGRAMAREA FUNCȚIONAL-PROCEDURALĂ A SISTEMELOR DE CALCUL CU RESURSE LIMITATE

Autori: Marin PODUBNII, Ghenadie SAFONOV, Constantin ABABII
Coordonator: conf., dr. V. Ababii

Universitatea Tehnică a Moldovei

Abstract: *Lucrarea „Programarea Funcțional-Procedurală a Sistemelor de Calcul cu Resurse Limitate” este dedicată elaborării unui mecanism pentru reducerea volumului de memorie ocupată de codul de program la implementarea algoritmilor bazați pe formule matematice complexe.*

Cuvinte cheie: *AVR, PIC, limbaje de programare, programare funcțională, programare procedurală, memoria pentru programe, topologia programului.*

1. Introducere

Programarea funcțională prezintă o paradigmă de programare unde calculul se tratează ca evaluarea unei funcții matematice. Aceste limbaje de programare sunt specifice mediilor academice, precum ar fi MatLab, Mathematica etc. [1]. Programarea procedurală prezintă o paradigmă de programare bazată pe conceptul de apel de procedură. Procedurile conțin o serie de pași care trebuie executați pentru a atinge o oarecare stare. Orice procedură poate fi apelată la orice moment de un program, de altă procedură, sau de ea însăși. Ca exemplu poate servi limbajul C/C++ cu succes utilizate în programarea dispozitivelor microcontroler [1].

Astăzi, practic este ne-imaginată funcționalitatea unui sistem, din diferite domenii ale activității umane, fără de utilizarea dispozitivelor microcontroler (MC). Practic zilnic apar noi sisteme bazate pe dispozitive microcontroler, care efectuează operații de calcul pentru diferite domenii de activitate umană, precum: automatizarea echipamentelor de uz casnic, casa inteligentă, sisteme robotice, comunicații etc. [1]. Totodată, complexitatea algoritmilor de procesare a datelor, devin din ce în ce tot mai mare, respectiv necesitând tot mai multe resurse de calcul și memorie pentru stocarea acestora. În aceste cazuri apare conflictul dintre calitatea serviciilor (care este oferită de algoritmul de procesare a datelor) și performanțele sistemului de calcul (care sunt oferite de dispozitivul microcontroler).

În scopul optimizării resurselor consumate de algoritmii de procesare a datelor, în lucrarea de față, se propune o metodă de programare funcțional-procedurală bazată pe organizarea topologică a memoriei de stocare a codului de program.

Tabelul 1. Caracteristicile unor produse oferite de companiile Atmel și MicroChip.

Dispozitivul MC	Memorie FLASH, Program	Memorie SRAM, Data
ATmega8	8 KB	1 KB
ATmega16	16 KB	1 KB
ATmega32	32 KB	2 KB
ATmega64	64 KB	4 KB
ATmega128	128 KB	4 KB
ATmega2560	256 KB	8 KB
PIC16C556	1024*14 W	96 B
PIC16C62	2048*14 W	128 B
PIC16C63	4096*14 W	192 B
PIC16C66	8192*14 W	368 B
PIC16C77	8192*14 W	368 B

2. Dispozitive MC cu resurse de calcul limitate

Pentru analiza comparativă a resurselor sau selectat dispozitive MC ale două companii, prezente cel mai frecvente pe piață. Acestea sunt compania Atmel cu produsele MegaAVR [2] și compania MicroChip cu produsele PIC16 [3]. În Tabelul 1 sunt prezentate caracteristicile unora din produsele oferite de aceste companii. Analiza parametrică a acestor dispozitive arată o prezență limitată de memorie de stocare a codului de program, ceea ce duce la limitarea domeniului de aplicare sau la reducerea complexității algoritmului de procesare a datelor, respectiv reducând și calitatea funcțională a sistemului implementat în baza acestor dispozitive MC.

3. Descrierea matematică a metodei de programare funcțional-procedurală

Fie definit algoritmul $A = \{a_i, i = \overline{1, N}\}$, unde a_i este mulțimea de funcții aritmetice, logice, trigonometrice etc. Pentru mulțimea de funcții A este stabilită regula:

$$a_i \cap a_j \neq \emptyset, \forall (i = \overline{1, N}, j = \overline{1, N}, \&i \neq j), \quad (1)$$

ceea ce indică faptul că, orice funcție $a_i, \forall i = \overline{1, N}$ poate să se repere de mai multe ori în algoritmul A .

Fie definit modelul matematic $M = \{m_l, l = \overline{1, L}\}$ care asigură $A \xrightarrow{M} P$, unde $P = \{p_k, k = \overline{1, K}\}$ este programul care realizează algoritmul A , $p_k, \forall k = \overline{1, K}$ este mulțimea de proceduri pentru realizarea algoritmului A , și $m_l, \forall l = \overline{1, L}$ este mulțimea de reguli pentru transformarea algoritmului A în programul P . În Figura 1 este prezentată structura procesului de transformare a algoritmului A în programul P . Programul P prezintă topologia spațială a amplasării mulțimii de proceduri $p_k, \forall k = \overline{1, K}$ și interacțiunea dintre acestea.

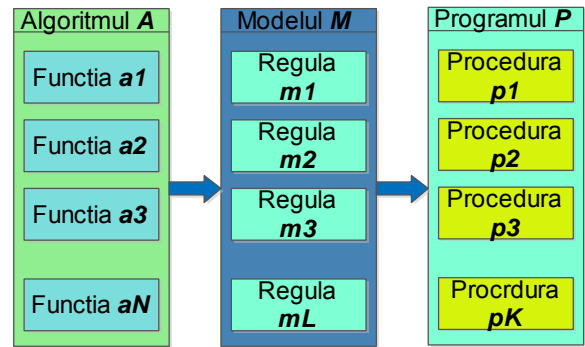


Fig. 1 Procesul de transformare.

4. Exemplu de programare funcțional-procedurală

Fie definit algoritmul:

$$A = \left\{ \left((x^2 + \cos(x))^2 + (\sin(y) * y^2)^2 \right)^2 + (2 * \sin(x * y))^2 \right\} \quad (2)$$

în rezultatul transformărilor vom obține mulțimea de proceduri:

$$P = \{QRT(x), COS(x), SIN(x), ADD(x_1, x_2), MUL(x_1, x_2)\}, \quad (3)$$

unde QRT, COS, \dots, MUL sunt numele procedurilor și x argumentul acestora. În

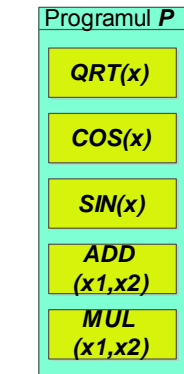


Fig. 2 Topologia programului.

Figura 2 este prezentată topologia programului P pentru realizarea algoritmului (2). Secvența de lansare a procedurilor din mulțimea (3), pentru calculul algoritmului (2) este prezentată în modelul (4):

$$ADD \left(\underbrace{QRT \left(\underbrace{ADD \left(\underbrace{QRT \left(\underbrace{ADD \left(\underbrace{QRT(x), COS(x)}_{1, 2} \right)}_{3} \right)}_{4, 5} \right)}_{6} \right)}_{7}, \underbrace{QRT \left(\underbrace{MUL \left(\underbrace{SIN(y), QRT(y)}_{4, 5} \right)}_{6} \right)}_{8} \right)}_{9}, \underbrace{QRT \left(\underbrace{MUL \left(\underbrace{2, SIN \left(\underbrace{MUL(x, y)}_{8} \right)}_{9} \right)}_{10} \right)}_{11} \right) \quad (4)$$

5. Concluzii

În lucrarea de față sunt prezentate rezultatele cercetărilor și testărilor efectuate în procesul de elaborare a produselor program pentru sisteme de calcul bazate pe dispozitive microcontroler cu resurse limitate de stocare a codului de program. În rezultatul aplicării metodei dezvoltate în lucrare s-a obținut o reducere de $\approx 5\%$ a volumului de spațiu ocupat de program la implementarea algoritmului definit în modelul matematic (2).

Rezultatele obținute în această lucrare vor fi aplicate în continuare pentru implementarea sistemelor de control în timp real a roboților mobili și pentru implementarea algoritmilor pentru sisteme multi-agent.

Mențiuni

Lucrarea de față a fost elaborată în cadrul Centrului de Creativitate Tehnică „Hard and Soft” a Universității Tehnice a Moldovei.

Bibliografie

1. <http://ro.wikipedia.org/wiki/>, accesat 5.10.2013.
2. <http://www.atmel.com>, accesat 10.11.2013.
3. <http://www.microchip.com>, accesat 13.11.2013.