

DUŠAN HRUBÝ, VLADIMÍR CVIKLOVIČ

Jazyk C a mikrokontroléry Silabs

Nitra 2016

Názov publikácie: Jazyk C a mikrokontroléry Silabs

Autori: prof. Ing. Dušan Hrubý, PhD.
Slovenská poľnohospodárska univerzita v Nitre,
Technická fakulta,
Katedra elektrotechniky, automatizácie a informatiky

Ing. Vladimír Cviklovič, PhD.
Slovenská poľnohospodárska univerzita v Nitre,
Technická fakulta,
Katedra elektrotechniky, automatizácie a informatiky

Recenzenti: doc. RNDr. Monika Božiková, PhD.
Slovenská poľnohospodárska univerzita v Nitre,
Technická fakulta,
Katedra fyziky

Ing. Marian Amrich, PhD.
Bel Power Solutions, s.r.o.,
Areál ZŤS 924,
018 41 Dubnica nad Váhom

Schválil rektor Slovenskej poľnohospodárskej univerzity v Nitre
dňa 7. 12. 2015 ako vysokoškolskú učebnicu.

© prof. Ing. Dušan Hrubý, PhD., Ing. Vladimír Cviklovič, PhD.

ISBN 978-80-552-1458-0

OBSAH

POJMY, SKRATKY A OZNAČENIA.....	5
ÚVOD.....	11
1. MIKROKONTROLÉRY 8051.....	13
1.1. Úvod do architektúry mikrokontrolérov.....	13
1.2. Intel.....	14
1.3. Atmel.....	15
1.4. Silicon Laboratories.....	17
1.5. Technické a programové prostriedky na vývoj programu.....	19
1.5.1. Výber programovacieho jazyka.....	19
1.5.2. Výber vývojového prostredia.....	21
1.5.3. Integrované vývojové prostredie a ladenie programu.....	22
2. PROGRAMOVANIE V JAZYKU SYMBOLICKÝCH ADRES.....	24
2.1. Pamäťové oblasti mikrokontroléra 8051.....	25
2.2. Spôsoby adresovania.....	26
2.3. Programovací jazyk JSA.....	29
2.3.1. Proces vývoja programu.....	30
2.3.2. Zdrojový program v JSA.....	32
2.3.3. Formát zdrojového programu v JSA.....	32
2.3.4. Všeobecné pravidlá zápisu programu.....	33
2.3.5. Pseudoinštrukcie prekladača (direktívy prekladača).....	33
2.3.6. Aktualizácia prekladača na nové klony 8051.....	35
2.4. Inštrukčný súbor ASM51.....	36
2.5. Príklady programov v JSA (ASM51).....	40
3. PROGRAMOVACÍ JAZYK C.....	44
3.1. História jazyka C.....	44
3.2. Jazyk ANSI C.....	45
3.2.2. Kľúčové slová a identifikátory.....	48
3.2.3. Operand, operátor, výraz.....	48
3.2.4. Terminálový vstup a výstup.....	50
3.2.5. Iteračné príkazy – cykly.....	52
3.2.6. Funkcie.....	54
3.2.7. Oblasť platnosti identifikátorov.....	55
3.2.8. Smerníky a polia.....	57
3.2.9. Odvođené a štruktúrované typy údajov.....	57
3.2.10. Preprocesor.....	59
3.2.11. Knižničné funkcie.....	59
4. PROGRAMOVACÍ JAZYK C51.....	62
4.1. Problémy s implementáciou.....	62
4.2. Odlišnosti jazyka C51 a C (ANSI).....	63
4.2.1. Údajové typy.....	63
4.2.2. Údajové oblasti.....	64
4.2.3. Pamäťové modely.....	65
4.2.4. Používané číselné formáty údajov.....	66
4.2.5. Prerušená.....	67
4.2.6. Obmedzenie kompilátora.....	68
4.2.7. Štandardné funkcie.....	68
5. KEIL C51.....	70
5.1. Vlastnosti.....	70
5.2. uVision4 IDE - Integrované vývojové prostredie Keil.....	71
5.2.1. Vytvorenie projektu.....	72
5.2.2. Odladovanie programu.....	76
5.2.3. Súčasti IDE.....	80
5.2.4. Demo verzia.....	80
5.3. Preklad jazyka C51 do JSA.....	81
5.4. Optimalizácia programu v jazyku C51.....	83
5.5. Migrácia na iný mikrokontrolér.....	85
5.6. Príklady v jazyku KEIL C51.....	87
6. FUNKCIE ZO ŠTANDARDNÝCH KNÍŽNÍC PODPOROVANÝCH V JAZYKU C51.....	92

6.1. Funkcie knižnice stdio.h	92
6.2. Funkcie knižnice stdlib.h	95
6.3. Funkcie knižnice math.h	97
6.4. Knižnica na prácu so znakmi ctype.h	99
6.5. Knižnica na prácu s reťazcami string.h	99
7. NULOVANIE, NAPÁJANIE A PRIPOJENIE K PROGRAMÁTORU	105
7.1. Možnosti nulovania mikrokontrolérov Silabs	105
7.2. Watchdog	107
7.3. Napájanie a pripojenie k programátoru mikrokontrolérov Silabs	109
7.4. Správa prerušení mikrokontrolérov	111
8. PORTY A ICH KONFIGURÁCIA	113
8.1. Režimy vývodov mikrokontroléra	113
8.2. Použitie mikrokontrolérov v 5-voltových systémoch	117
9. ZDROJE SYSTÉMOVEJ FREKVENCIE	119
9.1. Interný vysokofrekvenčný oscilátor	120
9.2. Interný nízko-frekvenčný oscilátor	121
9.3. Externý oscilátor	121
9.4. Násobič frekvencie	124
10. ČÍSLICOVÉ OBVODY MIKROKONTROLÉROV	126
10.1. Časovače	126
10.1.1. Štandardné časovače Timer0 a Timer1	126
10.1.2. Časovače Timer2 a Timer3	130
10.2. Programovateľné pole počítadiel	133
10.2.1. Režim zachytenia hrany impulzu	134
10.2.2. Režim softvérového časovača	135
10.2.3. Režim frekvenčného výstupu	137
10.2.4. Režim 8-bitového PWM výstupu	138
10.2.5. Modul 16-bitového PWM generátora	140
10.2.6. Používanie prerušení modulu PCA	141
11. ANALÓGOVÉ OBVODY MIKROKONTROLÉROV	143
11.1. Komparátory	143
11.2. AD prevodníky	145
11.2.1. Nastavenie a interpretácia nameraných údajov	145
11.2.2. Porovnávacia logika AD prevodníkov od Silabs	149
11.3. DA prevodníky	151
12. KOMUNIKAČNÉ ROZHRANIA MIKROKONTROLÉROV	153
12.1. UART	153
12.2. SPI	156
12.3. SMBus / I ² C	159
13. ZÁVER	165
14. LITERATÚRA	166

POJMY, SKRATKY A OZNAČENIA

0x00	zápis čísla v hexadecimálnom tvare
1011b	zápis čísla v binárnom tvare
63d	zápis čísla v dekadickom tvare
AD	analogovo-číslcový (Analog to Digital)
ADC	analogovo-číslcový prevodník (Analog to Digital Converter)
AMUX	analogový multiplexor
DA	číslcovo-analogový (Digital to Analog)
DAC	číslcovo-analogový prevodník (Digital to Analog Converter)
adresa	číslo označujúce miesto slabiky v pamäti alebo vstupno/výstupného portu, s ktorým chceme pracovať (tzn. kam chceme zapísať, odkiaľ chceme čítať). Maximálna veľkosť adresy určuje veľkosť adresového priestoru, teda počet slabík v pamäti alebo počet vstupno/výstupných portov
aktívny výstup	je schopný spínať ku kladnému aj zápornému pólu, pozostáva z dvojice komplementárnych tranzistorov, push-pull
ALU	aritmeticko-logická jednotka. Časť mikroprocesora, je to sčítačka doplnená o posuvné registre a logické obvody. Vykonáva operácie spojené so spracovaním údajov: matematické, logické a posuvy (rotácie). Počet bitov, s ktorými je schopná ALU pracovať, udáva, koľkobitový je mikroprocesor
ARM	architektúra rozvinutých mikrokontrolérov (60 % mobilných zariadení obsahuje ARM), vývoj začal vo firme ARM Holding
algoritmus	predpis, ktorý vedie od meniteľných vstupných informácií jednoznačne k požadovaným výsledkom v konečnom počte krokov
ASCII	americký štandardný kód na výmenu informácií (American Standard Code of Information Interchange)
assembler	prekladač jazyka symbolických adries, niekedy priamo označovaný ako JSA
bit	najnižšia jednotka nesúca informáciu, môže nadobúdať hodnoty 1 alebo 0
blok	časť zdrojového súboru, ktorý obsahuje deklarácie, definície a príkazy uzavreté v zložených zátvorkách
BUS	zbernica signálov rovnakého druhu použitia
C2	sériové, dvojvodičové rozhranie na programovanie mikrokontrolérov Silabs
C51	jazyk C určený na architektúru 8051
CLK	hodiny (clock), taktovanie obvodu
clock divider	delič frekvencie
CMOS	technológia výroby nízkoprikonových obvodov
counter	počítadlo (čes. čítač)
cyklus	slúži na opakovaný výpočet jedného príkazu alebo zloženého príkazu
datasheet	technický list súčiastky
debugger	všeobecne nástroj na odladovanie programu
definícia	premenné alebo funkcie vytvárajú spojenie medzi konkrétnou premennou, funkciou alebo údajovým typom a ich atribútami
deklarácia	premenné alebo funkcie vytvárajú rovnaké spojenie ako definícia, ale aj vymedzuje potrebné miesto v pamäti
dekrementácia	zníženie hodnoty premennej o jednu
EEPROM	reprogramovateľná pamäť
emulátor	zariadenie na hardvérové odladovanie programu
EPROM	pamäť nulovaná (mazateľná) ultrafialovým svetlom

Escape sekvencie (riadiace znaky) – dvojznak zložený zo znaku opačnej lomky \ a ďalším znakom; tiež sa označujú ako riadiace znaky, lebo sa nimi môžu zadávať príkazy na pohyb kurzora po zobrazovači	
FIFO	zásobník Queue, údaje odchádzajú v rovnakom poradí ako prišli (First In First Out)
firmvér	softvér mikrokontroléra, mikroprocesora
flag	indikačný bit
flash	rýchla reprogramovateľná pamäť
formálny parameter	– súčasť definície funkcie, ktorej hlavičkou sú formálne parametre uvedené v zátvorke. Formálne parametre sú určené údajovými typmi a ich menami a nad nimi sa vykonávajú príslušné operácie uvedené vo vnútri funkcie. Formálne parametre sú pri volaní funkcie prepísané hodnotami skutočných, aktuálnych parametrov
funkčný prototyp (prototyp funkcie) – je určený na deklaráciu funkcie a je vždy ukončený bodkočiarkou	
GND	napájacia a signálová zem
HCMOS	technológia výroby nízkoprikonových mikrokontrolérov
hlavička funkcie	je určená definíciou funkcie a je to z hľadiska syntaxe funkčný prototyp bez bodkočiarky
hlavičkový súbor	– slúži na doplnenie zdrojového súboru o funkcie, ktoré uložené v knižnici hlavičkových súborov, napr. <i>stdio.h</i> , <i>math.h</i> , <i>string.h</i>
I/O, V/V, PORT, vývod	– vstupno/výstupné rozhrania mikrokontrolérov
I²C (IIC)	komunikačná zbernica
IDE	integrované vývojové prostredie
identifikátor	pridelenie mena premennej, funkcie, štruktúry atď. Je dôležité dodržiavať veľké a malé písmená
inkrementácia	zvýšenie hodnoty premennej o jednu
INT	prerušenie programu mikrokontroléra (Interrupt)
inštrukcia	je príkaz v strojovom kóde, ktorý môže vykonávať počítač bez použitia prekladača. Každá inštrukcia obsahuje kód operácie, odkaz na použité operandy a odkaz na nasledujúcu inštrukciu, ktorou má program pokračovať. Inštrukcia je teda predpis na vykonanie určitej elementárnej operácie realizovateľnej priamo technickým vybavením počítača
inštrukčný cyklus	– je čas potrebný na výkon inštrukcie. Skladá sa z viacerých fáz, ich optimalizáciou je možné podstatnou mierou zrýchliť činnosť mikrokontroléra
ISP	sériové programovanie mikrokontroléra (Interface for in-System Programming)
JSA	jazyk symbolických adres
JTAG	emulačný prostriedok vyššej úrovne, komunikačné rozhranie pre programovanie (Join Test Action Group, asociácia založená v r. 1985)
kľúčové slovo	prvok množiny slov, ktorým prekladač programovacieho jazyka rozumie a má vo svojej slovnej zásobe
kompilátor	prekladač, programové vybavenie, ktoré zabezpečuje preklad zdrojového kódu zapísaného vo vyššom programovacom jazyku (JSA, C a pod.) do strojového kódu, ktorý môže procesor spracovať. Všeobecne je to program umožňujúci transformovať algoritmus zapísaný v textovom tvare do strojového kódu mikroprocesora
LIFO	zásobník, v ktorom údaje odchádzajú v opačnom poradí ako prišli, Stack (Last In / First Out)

log 0	logická nula
log 1	logická jednotka
LSB	najnižší bit v skupine bitov (Last Significant Bit)
mikrokontrolér	(jednočipový mikro počítač, monolitický mikro počítač, mikroriadič) pod všetkými týmito názvami je ukrytý mikro počítač štandardnej štruktúry vytvorený na jednej kremíkovej podložke v jedinom puzdre, vybavený pamäťami, perifériami a zbernicami tak, aby bol schopný autonómnej činnosti bez ďalších podporných obvodov, teda počítač, ktorý má všetky základné časti integrované v jednom puzdre. Mikrokontroléry sa používajú ako riadiace prvky v bežných priemyslových výrobkoch (napríklad domáce spotrebiče, spotrebná elektronika a pod.). Aj keď sú konštrukčne pomerne zložité, je ich cena vzhľadom na značnú sériovosť výroby nízka. Pri realizácii i jednoduchých riadiacich funkcií býva ich použitie často lacnejšie než realizácia tej istej funkcie z jednoduchších elektronických súčiastok
mikroprocesor	označujeme ním miniatúrny integrovaný obvod schopný spracovávať údaje na základe riadiacich inštrukcií tvoriacich program, je základnou riadiacou a výkonnou jednotkou nazývanou aj centrálna procesorová jednotka. Mikroprocesor je možné pomocou vonkajších vývodov doplniť o potrebné dodatočné obvody (pamäte a iné), čím poskytuje možnosť návrhu systému podľa potrieb
mikro počítač	(mikro počítačový systém) zariadenie, ktoré vznikne, ak k mikroprocesoru pridáme pamäťové, vstupno/výstupné a ďalšie obvody tak, aby bolo schopné vykonávať funkciu ako počítač, teda spracovávať informácie podľa zadaného programu. Ak toto zariadenie je tvorené pomocou modulov, nazývame ho pod pojmom „mikro počítačová stavebnica“
missing clock detector	– detektor chýbajúcej frekvencie, indikuje zlyhanie oscilátora
MIPS	milión inštrukcií za sekundu (Milion Instructions Per Second), aj architektúra procesorov, ktorých vývoj začal na Stanford University
MMU, MCU, CPU	– centrálna procesorová jednotka
MSB	najvyšší bit v skupine bitov (Most Significant Bit)
MUX	multiplexor
návratová hodnota	– každá funkcia vracia výsledok svojej činnosti, t. j. návratovú hodnotu. Pokiaľ o ňu nie je záujem, používame v jazyku C kľúčové slovo void (neplatný, prázdny)
obvody riadenia	časti mikroprocesora, zaisťujú vykonanie inštrukcie vytvorením postupnosti impulzov, ktoré ovplyvnia jednotlivé časti procesora tak, aby po ukončení tejto postupnosti bola inštrukcia vykonaná. Táto postupnosť je ovplyvnená mikroprogramom opisujúcim jednotlivé inštrukcie
OC	otvorený kolektor, pasívny výstup (Open Collector)
OTP	pamäť ROM s programovacou maskou pri výrobe čipu (One Touch Memory)
pamäť	časť mikroprocesorového systému, kde je uložený program alebo údaje
pasívny výstup	výstup v zapojení otvoreného kolektora, môže byť aj vstupom
periférie	všetky funkčné jednotky, ktoré sa nezúčastňujú priamo na vlastnom spracovaní informácií v rámci CPU, ale zabezpečujú ďalšie možnosti mikroprocesorového systému. Možno ich rozdeliť na integrované periférie, ktoré sú súčasťou puzdra mikrokontroléra (počítadlá a časovače, komunikačné rozhrania, programová a údajová pamäť, vstupno/výstupné brány) a vonkajšie periférie, ktoré sú tvorené samostatnými elektronickými

	obvodmi alebo zariadeniami (externé prevodníky, komunikačné zariadenia, klávesnice, zobrazovače, ovládače na pohony, inteligentné snímače).
pipeline	zreťazené spracovanie údajov
PLCC	druh päťice integrovaného obvodu
počítač	všeobecne označujeme stroj na spracovanie informácií na základe vopred zadaného programu
pointer	(ukazovateľ, smerník) každý objekt (premenná alebo funkcia je uložený v pamäti počítača na presne určenej adrese a zaberá definovaný počet bajtov podľa údajového typu. Pointer predstavuje adresu objektu v pamäti, je teda premennou, ktorej obsahom je adresa príslušného objektu v pamäti. Dá sa takisto povedať, že pointer predstavuje adresu bunky pamäte, v ktorom je uložená hodnota objektu, s ktorou sa bežne počíta
príkaz	písomný záznam algoritmu vo vyššom programovacom jazyku vykonávame pomocou postupnosti príkazov. Príkaz procesor vykoná po predchádzajúcom preložení programu kompilátorom
program	je postupnosť konečného počtu krokov vedúceho k vyriešeniu úlohy. Býva zvyčajne zapísaný v nejakom programovacom jazyku, ktorý je pre človeka (programátora) zrozumiteľný. Je postupnosťou inštrukcií, ktoré vedú k vykonaniu úlohy. Program je väčšinou uložený na disku vo forme súboru (typu EXE, COM). V ňom je uložený rad čísiel, ktoré znamenajú jednotlivé inštrukcie (strojový kód). Po spustení je buď celý program, alebo jeho časť uložená do pamäte počítača
programátor	zariadenie na naprogramovanie integrovaného obvodu, napáľovacie zariadenie; človek, ktorý programuje
push-pull	aktívny výstup mikrokontroléra
PWM	impulzová šírková modulácia
registre	časti mikroprocesora – sú rýchle pamäte určené na zaznamenávanie údajov a adries. Jednotlivé mikroprocesory sa od seba líšia počtom registrov a ich veľkosťou, ktorá udáva, aké veľké číslo sme schopní v ňom uchovať. Ak register slúži ako vstupný a výstupný na hodnoty určené ALU, nazýva sa akumulátor
reset	vývod procesora na spustenie programu od adresy 0, nulovanie
reset value	hodnota v statických funkčných registroch po nulovaní
reťazcový literál	– je množina znakov zapísaná do úvodzoviek a ukončených \0 (platí pre jazyk C)
ROM	pamäť len s možnosťou čítania údajov
RS232C	štandardné komunikačné rozhranie
RTC	jednotka reálneho času, obvod reálneho času
SAR	AD prevodník s postupnou aproximáciou, aproximačný register
SRF	statický funkčný register
Silabs	Silicon Laboratories – výrobca mikrokontrolérov
simulátor	softvérový prostriedok na odľadovanie programu
skutočný parameter	– pri volaní funkcie sa hodnota formálneho parametra v hlavičke funkcie prekopíruje (nahradí, zamení) hodnotou skutočného (aktuálneho) parametra.
slabika	bajt, pölslovo, označenie pre 8 bitov (bity číslujeme 7..0), môže niesť hodnotu čísla so znamienkom (-128-127, shortint) alebo bez znamienka (0-255, bajt); za počet slabík píšeme B (kB, MB). angl. byte

slovo	word, dve slabiky, označenie pre 16 bitov (bity číslujeme 15..0 (7..0, 7..0)), môže niesť hodnotu čísla so znamienkom (-32768 až -32767, integer) alebo bez znamienka (napr. adresa, 0 až 65535). angl. word
SPI	sériová komunikačná zbernica (Serial Peripheral Interface) ozn. aj SSI
SYSCCLK	hodinová frekvencia mikroprocesora, mikrokontroléra (SYStem CLocK)
sysTÉM	množina prvkov a množina väzieb medzi nimi. Vykazuje voči svojmu okoliu vzťahy vstupu a výstupu a cieľové chovanie
telo funkcie	postupnosť príkazov uzatvorených medzi ľavou a pravou zloženou zátvorkou, kde pred ľavou zloženou zátvorkou je definícia funkcie s jej hlavičkou
timer	časovač
U_{cc}	kladné napájacie napätie
USB DEBUG ADAPTER	– emulátor firmy Silicon Laboratories
údaje	(dáta) informácie, ktoré majú formálny tvar vyhovujúci na spracovanie človekom alebo na strojové spracovanie
vstupno/výstupné porty	– sú obvody, ktoré sú určené na odovzdanie údajov medzi počítačom a okolím.
X51	všeobecne rad mikrokontrolérov 8051
XRAM	externá RAM mikroprocesora, môže byť aj na čípe mikrokontroléra
zásobník	(stack, queue) časť pamäte slúžiaca na odkladanie údajov, prípadne na odovzdanie hodnôt medzi podprogramami
zdrojový kód	zápis algoritmu riešenej úlohy vytvorený programátorom v príslušnom programovacom jazyku, napr. v Asembleri, C
zložený príkaz	tvorený postupnosťou jednoduchých príkazov uzavretých do zložených zátvoriek {}. Na rozdiel od bloku neobsahuje deklarácie a definície.

Poznámky autorov:

- časti programového kódu sú zobrazené vo fonte **Courier New, Bold**,
- niektoré anglické pojmy, ktoré ešte nezdomácnili, sú v učebných textoch nepreložené alebo preložené len približne,
- v obrázkoch opisu registrov sú bity určené len na čítanie vyplnené jemne sivou farbou,
- sú predpokladané vedomosti z terminológie z informačných systémov, základov elektrotechniky a elektroniky.

ÚVOD

Vysokoškolská učebnica je zameraná na programovanie mikrokontrolérov. Bol vybraný programovací jazyk C51, ktorý sa javí v súčasnej dobe ako veľmi efektívny a perspektívny v oblasti 8-bitových mikrokontrolérov. Jazyk symbolických adries (skrátene JSA, programátorskou komunitou nazývaný aj Asembler) je však rovnako potrebný na dôkladnejšie pochopenie väzby medzi jazykom C51 a technickými prostriedkami mikrokontrolérov. Na ilustráciu bola vybraná rodina mikrokontrolérov firmy Silicon Laboratories. Vzhľadom na ich progresívne vlastnosti, bohatý výber typov a prepracované vývojové nástroje sú veľmi vhodné aj pedagogické účely.

O čom sa vo vysokoškolskej učebnici môžete dočítať?

Odsek „Mikrokontroléry 8051“ hovorí o základných encyklopedických vedomostiach v tejto oblasti. Znalosti v odseku „Technické a programové prostriedky na vývoj programu“ sú potrebné na to, aby bolo možné vytvoriť si jednoduché vývojové pracovisko na pohodlnú prácu pri odlaďovaní navrhnutého systému.

Cieľom odseku „Programovanie v jazyku symbolických adries“ je zorientovať čitateľa v programovaní mikrokontrolérov pomocou základných inštrukcií na "najnižšej" úrovni a oboznamuje s inštrukčným súborom mikrokontrolérov 8051 v tomto jazyku. V odseku „Príklady programov v JSA (ASM51)“ sú znalosti z predchádzajúcej kapitoly názorne využité. V ďalších odsekoch sú zhrnuté základné znalosti jazyka ANSI C. Sú tu opísané základné rozdiely a vlastnosti štandardného jazyka C a jazyka C51 pre mikrokontroléry. Sú tu porovnané jazyky JSA a C51 a hľadané ich vzájomné súvislosti. Medzi najvhodnejšie vývojové prostredia na programovanie C51 v súčasnej dobe patrí Keil C51, ktorý je stručne opísaný. Všeobecné príklady v jazyku C a C51 dopĺňujú tieto teoretické informácie.

Odsek „Funkcie zo štandardných knižníc podporovaných v jazyku C51“ je zameraný na vysvetlenie činnosti štandardných funkcií v jazyku C51. Na každú funkciu je uvedený aj príklad. Umožňuje to lepšie pochopenie prakticky zameraným čitateľom.

Úlohou ostatných odsekov je oboznámiť čitateľa s programovaním prístupu k hardvéru mikrokontrolérov. Blokové schémy a vzorové príklady sú opísané, čitateľ môže ľahko preniknúť do podstaty tejto skutočne nádhernej problematiky.

Veríme, že táto vysokoškolská učebnica bude dobrým pomocníkom pri rozširovaní vedomostí v oblasti využitia mikrokontrolérov a praktických návrhov mikroprocesorových systémov.