

Inhaltsverzeichnis

1	Multitasking, Kontext, Real-Time & Co.	11
1.1	Kontext	11
1.2	Zustand einer Task	12
1.3	Scheduler.....	13
1.3.1	Kooperatives Scheduling	13
1.3.2	Round Robin Scheduling.....	14
1.3.3	Präemptives Scheduling	14
1.4	Multitasking und Echtzeitfähigkeit.....	15
2	Experimentierumgebung.....	17
2.1	Hardware.....	17
2.2	Entwicklungsumgebung	18
2.3	Interface-Schaltung.....	18
3	AVR-Controller in C programmieren.....	19
3.1	Digitale Ein- und Ausgaben.....	19
3.2	Die serielle Schnittstelle nutzen	21
3.2.1	Die serielle Schnittstelle initialisieren	22
3.2.2	Daten senden und empfangen	22
3.2.3	Beispielprogramm: Zeichen vom Terminal lesen	23
3.3	Analoge Werte messen	25
3.3.1	Den Analog-Digital-Wandler initialisieren	25
3.3.2	Analoge Spannungswerte einlesen	26
3.3.3	Beispielprogramm: Messungen am Spannungsteiler	27
3.4	PWM	30
3.4.1	Timer2 als PWM konfigurieren.....	30
3.4.2	Beispielprogramm: Motordrehzahl einstellen	32
3.5	Interrupt.....	35
3.5.1	Timer-Interrupt: Systemuhr	35
3.5.2	Beispielprogramm: Uhrzeit und blinkende LED	36
3.5.3	Hardware-Interrupt.....	38
3.5.4	Beispielprogramm: Drehgeber auswerten	41
3.6	Datenflussdiagramm	43

4	Multitasking, die Erste: Die Minimalversion	45
4.1	Die Grundidee	45
4.2	Der praktische Einsatz	47
4.2.1	Drehzahlregler mit Tachogenerator und Hardware-PWM	47
4.3	Zusammenfassung	58
5	Multitasking, die Zweite: Ein Scheduler im Eigenbau	59
5.1	Die Grundidee	59
5.2	Modul RTC-Scheduler	63
5.3	Der praktische Einsatz	65
5.3.1	RTC-Drehzahlregler mit Tachogenerator	65
5.3.2	RTC-Drehzahlregler mit Drehgeber und Hardware-PWM	72
5.3.3	RTC-Drehzahlregler mit Drehgeber und Software-PWM	82
5.3.4	RTC-Drehzahlregler für zwei Motoren mit FIFO und Software-PWM	92
5.4	Ein Blick auf die Details	106
5.4.1	Task-Liste	106
5.4.2	Task-Synchronisation	108
5.4.3	FIFO	110
5.4.4	Scheduler	113
5.4.5	Speicherbedarf	114
5.5	Zusammenfassung	115
6	Multitasking, die Dritte: Kooperation ist gefragt	117
6.1	Die Grundidee: Adam Dunkels' Protothreads	118
6.2	Modul COS-Scheduler	120
6.3	Der praktische Einsatz	123
6.3.1	COS: Teste die Möglichkeiten	123
6.3.2	COS-Drehzahlregler mit Tachogenerator und Hardware-PWM	130
6.4	COS-Drehzahlregler mit Drehgeber und Hardware-PWM	138
6.4.1	COS-Drehzahlregler für zwei Motoren mit FIFO und Software-PWM	148
6.5	Ein Blick auf die Details	159
6.5.1	Protothreads: Makros machen's möglich	159
6.5.2	Task-Liste	163
6.5.3	Semaphoren	164
6.5.4	FIFO	166
6.5.5	Scheduler	170
6.5.6	Speicherbedarf	172
6.6	Zusammenfassung	173

7	Multitasking, die Vierte: Präemptives Tasking	175
7.1	Die Grundidee	175
7.1.1	Task-Kontext	177
7.1.2	Semaphoren: gemeinsam genutzte Ressourcen schützen	179
7.1.3	Ablaufinvariante Funktionen: »Reentrant Functions«.....	182
7.2	Modul RTOS-Scheduler	183
7.3	Der praktische Einsatz.....	187
7.3.1	RTOS: Teste die Möglichkeiten	187
7.3.2	RTOS-Drehzahlregler mit Tachogenerator und Hardware-PWM.....	196
7.3.3	RTOS-Drehzahlregler mit Drehgeber und Hardware-PWM.....	205
7.3.4	RTOS-Drehzahlregler für zwei Motoren mit Software-PWM.....	215
7.4	Ein Blick auf die Details	226
7.4.1	Task-Liste	227
7.4.2	Task-Kontext.....	228
7.4.3	Kontext retten und wiederherstellen	229
7.4.4	Initialer Kontext einer Task.....	231
7.4.5	Kontextwechsel	233
7.4.6	Semaphoren	237
7.4.7	Messagebox	240
7.4.8	Scheduler	241
7.4.9	Zeitverhalten	245
7.4.10	Speicherbedarf.....	249
7.5	Zusammenfassung	250
A	Anhang.....	251
A.1	Interface-Schaltung für zwei Gleichstrommotoren mit Drehgeber	251
A.2	Pin-Belegung des benutzten Evaluation-Boards	253
A.3	Softwaremodule und Beispielprogramme	254
	Literaturverzeichnis	257
	Stichwortverzeichnis	259