
Inhaltsverzeichnis

1	Einleitung und Methodik	1
2	Analyse	11
2.1	Prinzipielles Vorgehen bei einer Analyse	12
2.2	Die Anwendungsumgebung	13
2.2.1	Nutzer der Anwendung	13
2.2.2	Nutzungsszenarien	14
2.3	Analyse der Aufgabenstellung	16
2.3.1	Textanalyse	16
2.3.2	Analyse von Nutzungsszenarien	22
2.3.3	Analyse der Interaktionen mit der Umgebung	22
2.3.4	Anwendungsfalldiagramme	30
2.3.5	Ein-/ Ausgabenanalyse	32
2.4	Zusammenfassung	37
2.5	Übungen	39
	Literaturangaben	40
3	Entwurf	41
3.1	Aktivitätsdiagramme	41
3.2	Entwurf eines Algorithmus	45
3.2.1	Aktivitätsanalyse	46
3.2.2	Datenflussanalyse	53
3.3	Ausnahme- und Fehlerbehandlung	57
3.4	Ein Beispiel	61
3.5	Zusammenfassung	64
3.6	Übungen	66
	Literaturangaben	66
4	Information, Daten und Datenverarbeitung	67
4.1	Information und ihre Darstellung	67
4.2	Darstellung von Ganzzahlen	72

4.2.1	Duale Zahlen	72
4.2.2	Arithmetische Operationen mit Dualzahlen.	76
4.2.3	Zweikomplementzahlen	78
4.2.4	Hexadezimale Zahlen.	83
4.3	Nichtnumerische Informationen.	84
4.3.1	Der ASCII-Code.	84
4.3.2	Der Unicode	86
4.3.3	Farbcodierung	86
4.4	Datenverarbeitung	86
4.5	Zusammenfassung	91
4.6	Übungen	92
	Literaturangaben	93
5	Einführung in die Programmiersprache C.	95
5.1	Verarbeitungsprinzip eines Rechners	95
5.2	Basiselemente der Programmiersprache C.	99
5.2.1	Aufbau eines C-Programms	100
5.2.2	Aufbau von C-Funktionen	103
5.2.3	Elementare Datentypen	105
5.2.4	Benutzerdefinierte Datentypen.	107
5.2.5	Namen, Bezeichner und Werte.	110
5.2.6	Konstante vs. Variable vs. Werte	114
5.2.7	Operatoren	115
5.3	Bibliotheksfunktionen	122
5.4	Das erste Programm	126
5.5	Zusammenfassung	130
5.6	Übungen	131
	Literaturangaben	132
6	Alternativen	133
6.1	Entwurf alternativer Ausführungen	134
6.1.1	Kurze Einführung in die Aussagenlogik	134
6.1.2	Einführung in die Boolesche Algebra	136
6.1.3	Analyse der Aufgabenstellung	139
6.2	Implementierung von Alternativen.	148
6.2.1	Alternative 1 aus 2	149
6.2.2	Alternative 1 aus N.	150
6.2.3	Fallunterscheidungen	154
6.3	Beispiele.	157
6.4	Zusammenfassung	166
6.5	Übungen	167
	Literaturangaben	169

7	Schleifen	171
7.1	Eigenschaften von Schleifen	171
7.2	Analyse und Entwurf von Schleifen	174
7.3	Implementierung von Schleifen	183
7.3.1	Zählschleifen	184
7.3.2	Universalschleifen	186
7.3.3	Abbruch im Schleifenblock	193
7.3.4	Endlosschleifen	197
7.4	Verschachtelte Schleifen	198
7.4.1	Verschachtelte Zählschleifen	201
7.4.2	Verschachtelte Universalschleifen	206
7.5	Zusammenfassung	211
7.6	Übungen	212
	Literaturangaben	214
8	Datenstrukturen	215
8.1	Von Einzeldaten zu Datenstrukturen	215
8.2	Datenanalyse	220
8.3	Homogene Datenstrukturen	231
8.3.1	Grundlagen	231
8.3.2	Arithmetische Anwendungen	233
8.3.3	Bearbeitung von Texten	235
8.3.4	Tabellengesteuerte Programmierung	238
8.3.5	Mehrdimensionale Felder	243
8.4	Inhomogene Datenstrukturen	245
8.5	Verschachtelte Datenstrukturen	247
8.6	Zusammenfassung	249
8.7	Übungen	250
	Literaturangaben	252
9	Funktionen	253
9.1	Funktionsdiagramme	254
9.2	Entwurf von Funktionen	257
9.2.1	Weitere Aspekte zur Definition von Funktionen	258
9.2.2	Implementierungsaspekte	264
9.3	Sichtbarkeit von Bezeichnern	266
9.4	Lebensdauer von Variablen	269
9.5	Funktionen mit Parametern	271
9.5.1	Parameterübergabe: Call by Value	273
9.5.2	Parameterübergabe: Call by Pointer	275
9.6	Implementierungsregeln bei der Nutzung von Funktionen	285
9.7	Beispiele	286
9.8	Inline-Funktionen	295

9.9	Makros	296
9.10	Zusammenfassung	299
9.11	Übungen	299
	Literaturangaben	301
10	Dynamische Datenstrukturen	303
10.1	Allgemeine Zeigernutzung	305
10.2	Zeiger und Datenstrukturen	312
10.3	Dynamische Speicherverwaltung	315
10.4	Dynamische Erzeugung statischer Datenstrukturen	322
10.5	Datenstrukturen mit variabler Struktur	325
	10.5.1 Basisoperationen mit einfach verketteten Listen	329
	10.5.2 Anwendungsfälle mit einfach verketteten Listen	338
10.6	Zusammenfassung	340
10.7	Übungen	341
	Literaturangaben	343
11	Anpassbare Softwarestrukturen	345
11.1	Funktionszeiger	348
11.2	Callback-Funktionen	351
11.3	Konfigurierbare Softwarestrukturen	353
11.4	Tabellengesteuerte Programmierung mit Funktionszeigern	357
11.5	Zusammenfassung	360
11.6	Übungen	361
	Literaturangaben	362
12	Ein-/Ausgaben und Dateien	363
12.1	Einrichten und Löschen des Pufferspeichers	366
12.2	Transfers zwischen Anwendungsspeicher und Pufferspeicher	370
12.3	Transfers zwischen Pufferspeicher und externem Gerät	377
12.4	Fehler bei der Nutzung von Dateien	378
12.5	Ein Beispiel	379
12.6	Zusammenfassung	396
12.7	Übungen	397
	Literaturangaben	398
13	Strukturierung der Software	399
13.1	Module als Hilfsmittel zur Strukturierung	400
13.2	Strukturierung durch Gruppenbildung	402
13.3	Interaktionen zwischen Modulen	406
13.4	Header-Dateien	408
13.5	Berücksichtigung des Modulkonzepts in der Dokumentation	413

13.6	Zusammenfassung	416
13.7	Übungen	417
	Literaturangaben	418
14	Fehlerbehandlung	419
14.1	Begriffe und Definitionen	419
14.2	Fehlerquellen in Programmen	421
14.3	Fehlerbehandlung in C-Programmen	422
14.3.1	Fehlererkennung	423
14.3.2	Fehlermeldung	423
14.3.3	Fehlerreaktion	426
14.4	Fehlersuche	434
14.5	Zusammenfassung	437
14.6	Übungen	438
	Literaturangaben	438
15	Hardware und C	439
15.1	Ausgewählte Hardwarekomponenten und -eigenschaften	440
15.1.1	Elementare Grundfunktionen	440
15.1.2	Speicherbausteine	444
15.1.3	Der Hauptspeicher	445
15.1.4	Digitale Ein- und Ausgabe	450
15.1.5	Hardware-Erweiterungen für Interrupts	452
15.2	Das Speichermodell	457
15.2.1	Speicherklassen und Typqualifizierer in C	458
15.2.2	Definition und Initialisierung von Variablen	460
15.2.3	Speicherzugriff über absolute Adressen	461
15.2.4	Nutzung unterschiedlicher Speichertechnologien	464
15.3	Bitverarbeitung	464
15.3.1	Bitmanipulation über Masken	465
15.3.2	Bitfelder	471
15.4	Zusammenfassung	472
15.5	Übungen	472
	Literaturangaben	473
16	Systemsoftware	475
16.1	Ebenen und Schnittstellen in der Systemsoftware	476
16.2	Entwurf der Systemschnittstellen	481
16.3	Interrupt-Programmierung	484
16.3.1	Anforderungen an die Interrupt-Nutzung	484
16.3.2	Ereignisse und Anwendungen	488
16.3.3	Initialisierung der Interrupt-Hardware	491

16.3.4	Interrupt-Servicefunktion	492
16.3.5	Übergabe der Ereignisinformation an die Anwendung	494
16.4	Ein Beispiel: Realisierung eines Timers	501
16.4.1	Der prinzipielle Aufbau von Hardwaretimern	503
16.4.2	API-Entwurf für emulierte Timer.	505
16.4.3	Entwurf der Hardware Abstraction Layer	507
16.4.4	Realisierung der Hardware-Abstraktionsebene	508
16.4.5	Realisierung der API-Funktionen.	513
16.5	Zusammenfassung	513
16.6	Übungen	515
	Literaturangaben	516
17	Echtzeitprogrammierung	517
17.1	Eigenschaften von Echtzeitsystemen	518
17.2	Analyse und Entwurf von Steuerungssoftware	527
17.3	Zustandsdiagramme	534
17.4	Entwurf des Steueralgorithmus	541
17.4.1	Ziele der Steuerung	543
17.4.2	Ein-/Ausgangsanalyse der Steuerung	544
17.4.3	Entwurf des Steueralgorithmus	544
17.4.4	Ereignisanalyse	548
17.4.5	Reaktionsanalyse	550
17.5	Hierarchische Zustandsgraphen	552
17.5.1	Eigenschaften hierarchischer Zustände	552
17.5.2	Transitionen in hierarchischen Zuständen	554
17.6	Nebenläufige Zustandsgraphen	559
17.7	Festlegung der Taktperiode der Uhr.	560
17.8	Zusammenfassung	562
17.9	Übungen	563
	Literaturangaben	565
18	Implementierung von Echtzeitsoftware	567
18.1	Der Grundzyklus einer synchronen Steuerung	568
18.2	Realisierung von Zustandsgraphen	573
18.2.1	Realisierung einfacher Zustandsgraphen	574
18.2.2	Realisierung hierarchischer Zustandsgraphen	584
18.3	Realisierung von Zeitereignissen	590
18.4	Steuerung und Überwachung der Ein- und Ausgaben	593
18.5	Zusammenfassung	594
18.6	Übungen	595
	Literaturangaben	595
	Stichwortverzeichnis	597