
Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Entwurfsmethodik	7
1.2.1	Erfassen und simulieren	7
1.2.2	Beschreiben und synthetisieren	8
1.2.3	Spezifizieren, explorieren und verfeinern	9
1.3	Abstraktion und Entwurfsrepräsentationen	10
1.3.1	Modelle	10
1.3.2	Synthese	12
1.3.3	Optimierung	26
1.4	Übungen	28
2	Modellierung	37
2.1	Einleitung	37
2.2	Klassifikation von Modellen	39
2.3	Petri-Netzmodell	40
2.3.1	Dynamische Eigenschaften von Petri-Netzen	43
2.4	Kontrollflussmodelle	45
2.4.1	Endliche Automaten (FSMs)	45
2.4.2	Erweiterte Zustandsmaschinenmodelle	47
2.4.3	Hierarchische, nebenläufige Zustandsmaschinen	49
2.4.4	Statecharts	49
2.5	Datenflussmodelle	54
2.5.1	Datenflussgraphen	54
2.5.2	Markierte Graphen	55
2.5.3	Synchrone Datenflussgraphen (SDF)	56
2.6	Erweiterte Datenflussmodelle \otimes	60
2.6.1	Verklebungen in SDF-Graphen	60
2.6.2	Reduktion des SDF-Modells auf markierte Graphen	62
2.6.3	Zyklostatischer Datenfluss	63
2.6.4	Dynamische Datenflussmodelle	64

2.7	Strukturorientierte Modelle	66
2.7.1	Komponenten-Verbindungsdiagramm (CCD)	66
2.8	Heterogene Modelle	66
2.8.1	Kontroll-Datenflussgraphen (CDFGs)	68
2.8.2	Modellierung in SystemCoDesigner	74
2.9	Können Programmiersprachen mehr?	79
2.10	Literaturhinweise	81
2.11	Übungen	86
3	Synthese	93
3.1	Fundamentale Syntheseprobleme	93
3.1.1	Ablaufplanung	94
3.1.2	Allokation	96
3.1.3	Bindung	97
3.2	Implementierung	98
3.3	Problemklassen	99
3.4	Übungen	101
4	Ablaufplanung	103
4.1	Klassifikation von Ablaufplanungsproblemen	103
4.1.1	Statische und dynamische Ablaufplanung	103
4.1.2	Präemption	104
4.1.3	Datenabhängigkeiten	104
4.1.4	Ressourcenbeschränkungen	104
4.1.5	Periodische Ablaufplanung	104
4.2	Klassifikation von Algorithmen	105
4.3	Ablaufplanung ohne Ressourcenbeschränkungen	106
4.3.1	Der ASAP-Algorithmus	106
4.3.2	Der ALAP-Algorithmus	107
4.3.3	Ablaufplanung mit Zeitbeschränkungen	108
4.4	Ablaufplanung mit Ressourcenbeschränkungen	112
4.4.1	Erweiterte ASAP- und ALAP-Verfahren	113
4.4.2	Listscheduling	113
4.4.3	Force-directed scheduling	117
4.4.4	Ganzzahlige lineare Programmierung	121
4.5	Periodische Ablaufplanungsprobleme \otimes	124
4.5.1	Iterative Algorithmen	125
4.5.2	Iterative Ablaufplanung ohne Ressourcenbeschränkungen ..	132
4.5.3	Verfahren zur Reduktion von P_{\min}	138
4.5.4	Iterative Ablaufplanung mit Ressourcenbeschränkungen ...	143
4.5.5	ILP-Modell zur Ablaufplanung iterativer Algorithmen	143
4.5.6	Beispiel: FIR-Filter	145
4.6	Ressourcetypbindung \otimes	147
4.7	Dynamische Ablaufplanung	149
4.7.1	Definitionen	151

4.7.2	Optimierungskriterien bei dynamischen Ablaufplanungsproblemen	154
4.7.3	Dynamische Ablaufplanung ohne Echtzeitanforderungen ...	155
4.7.4	Dynamische Ablaufplanung mit Echtzeitanforderungen ...	160
4.7.5	Periodische, dynamische Ablaufplanung	165
4.8	Literaturhinweise, Ausblicke und Werkzeuge	170
4.9	Übungen	175
5	Bindung	181
5.1	Graphentheoretische Betrachtung von Bindungsproblemen	181
5.2	Bindung nach Ablaufplanung	185
5.3	Periodische Bindungsprobleme \otimes	188
5.4	Bindung hierarchischer Graphen \otimes	192
5.5	Bindung vor Ablaufplanung \otimes	192
5.6	Partitionierungsalgorithmen	193
5.6.1	Konstruktive Verfahren zur Partitionierung	195
5.6.2	Iterative Verfahren zur Partitionierung	198
5.6.3	Evolutionäre Algorithmen	199
5.6.4	Partitionierung mit linearer Programmierung	199
5.6.5	Algorithmen zur HW/SW-Partitionierung	200
5.7	Entwurfssysteme zur Partitionierung	202
5.7.1	Funktionale Partitionierung im Hardwareentwurf	203
5.7.2	ILP-basierte Ansätze für Multi-Chip-Entwürfe	205
5.7.3	Entwurfssysteme zur Hardware/Software-Partitionierung ...	206
5.7.4	ILP-basierte Ansätze zur HW/SW-Partitionierung	208
5.8	Literaturhinweise und Zusammenfassung	208
5.9	Übungen	210
6	Architektursynthese	217
6.1	Was ist Architektursynthese?	218
6.1.1	Operationswerk	219
6.1.2	Steuerwerk	221
6.2	Schätzung der Entwurfsqualität	223
6.2.1	Kostenmaße	224
6.2.2	Performanzmaße	224
6.2.3	Abschätzung der Taktperiode T	227
6.2.4	Abschätzung der Latenz L	229
6.2.5	Abschätzung der Ausführungszeit	229
6.2.6	Abschätzung der Kosten	229
6.3	Energie- und Leistungsverbrauchsmanagement	232
6.3.1	Energie- und Leistungsverbrauch	233
6.3.2	Leistungseffizienz	233
6.3.3	Quellen des Leistungsverbrauchs	236
6.3.4	Strukturelle Maßnahmen zur Leistungsverbrauchsreduktion.	237

6.3.5	Maßnahmen auf Verhaltensebene zur Reduktion des Leistungsverbrauchs	240
6.3.6	Schätzung des mittleren Leistungsverbrauchs	243
6.4	Notation und Problemstellung der Architektursynthese	246
6.5	Spezialoperationen und Spezialressourcen	250
6.5.1	Verkettung funktionaler Operatoren	250
6.5.2	Module mit Fließbandverarbeitung	252
6.5.3	Funktionale Fließbandverarbeitung und Schleifenfaltung	252
6.6	Verfahren zur Architektursynthese	255
6.6.1	Erweiterung bekannter Ablaufplanungsverfahren	255
6.6.2	ILP-Modelle zur Architektursynthese	255
6.7	ILP-Modelle zur Architektursynthese	257
6.7.1	Zeitschlitzmodelle	257
6.7.2	Flussmodell	262
6.8	Das Flussmodell	266
6.9	Analyse des Flussmodells \otimes	271
6.9.1	Polyedertheorie	271
6.9.2	Analyse des Lösungspolytops des Fluss-ILP	273
6.10	Verbesserung der Modellstruktur \otimes	280
6.10.1	Elimination von Flussvariablen	280
6.10.2	Anpassung der Konstanten $\delta(v_i, v_j, k)$	283
6.10.3	Elimination redundanter Ungleichungen	285
6.10.4	Hinzufügen gültiger Ungleichungen	286
6.11	Fließbandverarbeitung	288
6.11.1	Ressourcen mit Fließbandverarbeitung	288
6.11.2	Funktionale Fließbandverarbeitung und Schleifenfaltung	289
6.11.3	ASAP- und ALAP-Zeiten bei iterativer Ablaufplanung	292
6.11.4	Ablaufplanung mit vollstatischer Bindung \otimes	293
6.12	Relaxationsverfahren \otimes	296
6.13	Beispiele	297
6.13.1	Löser einer gewöhnlichen Differentialgleichung	297
6.13.2	Elliptisches Wellenfilter	299
6.14	Literaturhinweise und Werkzeuge	301
6.15	Übungen	309
7	Softwaresynthese	315
7.1	Merkmale und Klassifikation von Prozessoren	316
7.1.1	Klassifikation von Prozessoren	317
7.1.2	Kriterienbasierte Prozessorwahl	319
7.1.3	Architekturparameter von ASIPs	320
7.2	Einführung Compiler	322
7.2.1	Umgebungen	323
7.2.2	Sprachkonzepte	324
7.2.3	Analyse von Quellprogrammen	324
7.2.4	Phasen (Analyse, Codegenerierung, Optimierung)	328

7.3	Zwischendarstellungen	331
7.3.1	Drei-Adress-Code	331
7.3.2	Kontrollflussgraphen und DAGs	334
7.4	Codegenerierung und Codeoptimierung	338
7.4.1	Eine einfache Zielsprache	340
7.4.2	Allokation, Ablaufplanung und Bindung	341
7.5	Registervergabe und Registerbindung	346
7.5.1	Lebenszeitanalyse	346
7.5.2	Registervergabe und -bindung durch Graphfärbung	348
7.5.3	Globale Registervergabe	349
7.6	Optimale Registerbindung mit ILP-Techniken \otimes	352
7.6.1	Existierende Ansätze	352
7.6.2	Exakte minimale Färbung mit ILP-Modell	356
7.6.3	k -Färbung und Registerabwurf	358
7.6.4	Optimale k -Färbung mit ILP-Modell	359
7.6.5	Modellierung von Register-Register-Umspeicherungen	362
7.6.6	Bestimmung optimaler Registerabwurfzeitpunkte	365
7.6.7	Modellierung heterogener Registerstrukturen	368
7.6.8	Ausblicke	369
7.7	Codeselektion: Mustererkennung und Baumtransformation	370
7.8	Dynamische Programmierung	375
7.8.1	Prinzip der dynamischen Programmierung	376
7.8.2	Codegenerierung mit dynamischer Programmierung	377
7.9	Maschinenunabhängige Optimierung	380
7.9.1	Transformationen auf Grundblöcken	381
7.9.2	Globale Optimierung	383
7.10	Maschinenabhängige Optimierung	384
7.11	Softwaresynthese für eingebettete Prozessoren	386
7.11.1	Anpassbarkeit von Compilern	386
7.11.2	Unterschiede der Codegenerierung	387
7.11.3	Retargetierbare Compiler für RISC/CISC	388
7.11.4	Retargetierbare Compiler für DSPs	390
7.11.5	Retargetierbare Compiler für VLIW-artige Architekturen	393
7.11.6	Kommerzielle retargetierbare Compiler	400
7.12	Softwaresynthese auf Modulebene	401
7.12.1	Betriebssysteme	401
7.12.2	Echtzeitbetriebssysteme	403
7.12.3	Codegenerierung für datenflussdominante Systeme	404
7.12.4	Externe Ereignisse \otimes	412
7.12.5	Ressourcenzugriffsprotokolle und Ablaufplanungsanomalien \otimes	415
7.13	Literaturhinweise und Zusammenfassung	418
7.14	Übungen	422

8	Systemsynthese	427
8.1	Modell der Systemsynthese	430
8.1.1	Spezifikation eines Problems	431
8.1.2	Spezifikation von Zielarchitekturen	431
8.1.3	Spezifikation von Abbildungen	433
8.1.4	Implementierung	434
8.2	Entwurfsraumexploration	439
8.2.1	Optimierungsstrategien und Zielfunktionen	442
8.2.2	Schätzung der Entwurfsqualität	445
8.2.3	Strategien zur Überdeckung und zum Beschneiden des Entwurfsraums	447
8.3	Entwurfsraumexploration mit evolutionären Algorithmen	450
8.3.1	Optimierung mit evolutionären Algorithmen	452
8.3.2	Codierung von Implementierungen	454
8.3.3	Fitnessfunktionen und Beschränkungen	460
8.3.4	Parameter des evolutionären Algorithmus	463
8.3.5	Fallstudie	463
8.4	Modellerweiterungen	471
8.4.1	Mehrstufiger Spezifikationsgraph	471
8.4.2	Hierarchische Spezifikationsgraphen \otimes	473
8.5	Verbesserte Überdeckung des Entwurfsraums \otimes	479
8.5.1	Symbolische Techniken in der Entwurfsraumexploration ...	479
8.5.2	Pareto-Front-Arithmetik	485
8.6	Entwurfssysteme zur Systemsynthese	490
8.7	Zusammenfassung und Literaturhinweise	493
8.8	Übungen	498
Anhang	501
Notation	501
A.1	Logik	501
A.2	Mengen	502
A.3	Relationen und Funktionen	502
A.4	Lineare Algebra	503
A.5	Ganzzahlige Algebra	503
A.6	Graphen	503
A.6.1	Ungerichtete Graphen	504
A.6.2	Gerichtete Graphen	506
A.6.3	Perfekte Graphen	507
A.7	Polyedertheorie	508
A.8	Kombinatorische Optimierungsprobleme	510
A.8.1	Entscheidungsprobleme und Optimierungsprobleme	510
A.8.2	Algorithmen	510
A.8.3	Klassifikation von Problemen	511

Fundamentale Optimierungsalgorithmen	513
B.1 Lineare und ganzzahlige lineare Programmierung.....	514
B.1.1 Umformungen linearer Programme	515
B.1.2 Algorithmen zur Lösung von LPs	515
B.1.3 Algorithmen zur Lösung von ILPs	516
B.2 Simulated annealing	519
B.3 Evolutionäre Algorithmen	520
B.4 Mehrzieloptimierung.....	522
B.4.1 Leistungsbewertung von Optimierungsalgorithmen	523
B.4.2 Evolutionäre Algorithmen für die Mehrzieloptimierung	527
Graphenalgorithmen	531
C.1 Kürzeste- und Längste-Pfad-Probleme	531
C.1.1 KPP auf azyklischen Graphen: Topologische Sortierung	532
C.1.2 KPP auf zyklischen Graphen.....	532
C.1.3 LPP	537
C.1.4 Lösung von LPP (KPP) durch lineare Programmierung	537
C.2 Graphfärbung	538
C.2.1 Färbung triangulierter Graphen.....	539
C.2.2 Periodische Färbungsprobleme	541
C.3 Cliquepartitionierung	544
Literatur	547
Sachverzeichnis	571



<http://www.springer.com/978-3-540-46822-6>

Digitale Hardware/Software-Systeme

Synthese und Optimierung

Teich, J.; Haubelt, C.

2007, XV, 594 S., Softcover

ISBN: 978-3-540-46822-6