

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Motivation	1
1.2	Der Verifikationsprozess	11
1.2.1	Das V-Modell	13
1.2.2	Das Doppeldachmodell des Entwurfsprozesses	14
1.2.3	Das Doppeldachmodell des Verifikationsprozesses	18
1.3	Eine kurze Geschichte der Verifikation	22
1.4	Beispiele	29
1.5	Ausblick	34
1.6	Literaturhinweise	34
<b>2</b>	<b>Spezifikation digitaler Systeme</b>	<b>37</b>
2.1	Wie spezifiziert man ein System?	37
2.2	Formale Verhaltensmodelle	41
2.2.1	Petri-Netze	41
2.2.2	Endliche Automaten	47
2.2.3	Datenflussgraphen	51
2.2.4	Heterogene Modelle	56
2.3	Ausführbare Verhaltensmodelle	59
2.3.1	SystemC	60
2.3.2	SysteMoC	68
2.4	Formale Spezifikation funktionaler Anforderungen	72
2.4.1	Temporale Strukturen	73
2.4.2	Temporale Aussagenlogik	75
2.4.3	Die Zusicherungssprache PSL	83
2.5	Formale Spezifikation nichtfunktionaler Anforderungen	88
2.6	Literaturhinweise	91
<b>3</b>	<b>Verifikation</b>	<b>95</b>
3.1	Verifikationsaufgabe, -ziel und -methode	95
3.1.1	Verifikationsziel	97

3.1.2	Verifikationsmethode .....	99
3.2	Beobachtbarkeit und Steuerbarkeit .....	107
3.3	Gesteuerte zufällige Simulation .....	111
<b>4</b>	<b>Äquivalenzprüfung .....</b>	<b>115</b>
4.1	Implizite Äquivalenzprüfung .....	117
4.1.1	Kanonische Funktionsrepräsentationen .....	117
4.1.2	Taylor-Expansions-Diagramme .....	120
4.1.3	Reduktion und Normalisierung von TEDs .....	120
4.1.4	Kanonizität von TEDs .....	124
4.1.5	Implizite Äquivalenzprüfung mit TEDs .....	125
4.2	Explizite Äquivalenzprüfung .....	129
4.2.1	Regressionstest .....	131
4.2.2	Bereichstest .....	133
4.2.3	Pfadbereichstest .....	134
4.2.4	Fehleroffenbarende Unterbereiche .....	139
4.3	Sequentielle Äquivalenzprüfung .....	141
4.3.1	Automaten-Äquivalenz .....	142
4.3.2	Zustandsraumtraversierung .....	144
4.3.3	Symbolische Zustandsraumtraversierung .....	148
4.3.4	Erzeugung von Gegenbeispielen .....	151
4.4	Strukturelle Äquivalenzprüfung .....	152
4.5	Literaturhinweise .....	154
<b>5</b>	<b>Eigenschaftsprüfung .....</b>	<b>155</b>
5.1	Prüfung funktionaler Eigenschaften .....	156
5.1.1	Eigenschaftsprüfung auf Erreichbarkeitsgraphen .....	158
5.1.2	Strukturelle Eigenschaftsprüfung von Petri-Netzen .....	167
5.1.3	Partialordnungsreduktion .....	172
5.2	Explizite Modellprüfung .....	178
5.2.1	CTL-Modellprüfung .....	179
5.2.2	LTL-Modellprüfung .....	185
5.2.3	Zusicherungsbasierte Eigenschaftsprüfung .....	188
5.3	Symbolische Modellprüfung .....	197
5.3.1	BDD-basierte CTL-Modellprüfung .....	197
5.3.2	SAT-basierte Modellprüfung .....	199
5.4	Prüfung nichtfunktionaler Eigenschaften .....	207
5.4.1	Zeitbehaftete Petri-Netze .....	207
5.4.2	Zeitbehaftete Automaten .....	214
5.4.3	Zeitbehaftete SDF-Graphen .....	222
5.5	Literaturhinweise .....	232

<b>6</b>	<b>Hardware-Verifikation</b>	235
6.1	Äquivalenzprüfung kombinatorischer und sequentieller Schaltungen	236
6.1.1	Implizite Äquivalenzprüfung auf der Logikebene	236
6.1.2	Explizite Äquivalenzprüfung auf der Logikebene	246
6.1.3	Formale explizite Äquivalenzprüfung von Schaltwerken	258
6.1.4	Strukturelle Äquivalenzprüfung auf der Logikebene	263
6.2	Äquivalenzprüfung arithmetischer Schaltungen	273
6.2.1	Implizite Äquivalenzprüfung auf der Architekturebene	273
6.2.2	Äquivalenzprüfung zwischen Architektur- und Logikebene	280
6.2.3	Äquivalenzprüfung auf der Architekturebene	283
6.3	Formale Verifikation von Prozessoren	291
6.3.1	Äquivalenzprüfung für Prozessoren mit Fließbandverarbeitung	293
6.3.2	Berücksichtigung von Multizyklen-Funktionseinheiten, Ausnahmebehandlung und Sprungvorhersage	308
6.3.3	Äquivalenzprüfung für Prozessoren mit dynamischer Instruktionsablaufplanung	313
6.4	Funktionale Eigenschaftsprüfung	323
6.4.1	Zusicherungsbasierte Eigenschaftsprüfung	323
6.4.2	SAT-basierte Modellprüfung	331
6.5	Zeitanalyse	345
6.5.1	Zeitanalyse synchroner Schaltungen	345
6.5.2	Zeitanalyse latenzinsensitiver Systeme	351
6.6	Literaturhinweise	356
<b>7</b>	<b>Software-Verifikation</b>	361
7.1	Formale Äquivalenzprüfung eingebetteter Software	362
7.1.1	Äquivalenzprüfung von Assemblerprogrammen	362
7.1.2	Strukturelle Äquivalenzprüfung von Assemblerprogrammen	368
7.1.3	Äquivalenzprüfung von C-Programmen	373
7.2	Testfallgenerierung zur simulativen Eigenschaftsprüfung	391
7.2.1	Funktionsorientierte Testfälle	391
7.2.2	Kontrollflussorientierte Testfälle	400
7.2.3	Datenflussorientierte Testfälle	410
7.3	Formale funktionale Eigenschaftsprüfung von Programmen	416
7.3.1	Statische Programmanalyse	416
7.3.2	SAT-basierte Modellprüfung von C-Programmen	422
7.3.3	Modellprüfung durch Abstraktionsverfeinerung	425
7.4	Zeitanalyse	431
7.4.1	BCET- und WCET-Analyse	432
7.4.2	Echtzeitanalyse für Einprozessorsysteme	438
7.5	Literaturhinweise	448

<b>8</b>	<b>Systemverifikation</b>	451
8.1	Funktionale Eigenschaftsprüfung von SystemC-Modellen	452
8.1.1	Symbolische CTL-Modellprüfung von SysteMoC-Modellen	452
8.1.2	Modellprüfung von SystemC-Modellen	466
8.1.3	Formale Modellprüfung von Transaktionsebenenmodellen	476
8.1.4	Zusicherungsbasierte Eigenschaftsprüfung für Transaktionsebenenmodelle	484
8.2	Zeitanalyse auf Systemebene	490
8.2.1	Simulative Zeitbewertung	492
8.2.2	Kompositionale Zeitanalyse über Ereignisströme	499
8.2.3	Modulare Zeitanalyse mit RTC	508
8.3	Literaturhinweise	520
	<b>Anhang</b>	523
	<b>Notation</b>	523
A.1	Mengen	523
A.2	Relationen und Funktionen	524
A.3	Aussagenlogik	527
A.4	Prädikatenlogik erster Ordnung	528
A.5	Graphen	529
	<b>Binäre Entscheidungsdiagramme</b>	533
B.1	Entscheidungsdiagramme	533
B.2	Binäre Entscheidungsdiagramme	534
B.3	Verallgemeinerte binäre Entscheidungsdiagramme	537
	<b>Algorithmen</b>	541
C.1	Klassifikation von Algorithmen	541
C.2	SAT-Solver	542
C.3	SMT-Solver	551
C.4	CTL-Fixpunktberechnung	556
	<b>Literatur</b>	561
	<b>Sachverzeichnis</b>	587