

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Allgemeine Problemstellung . . . . .	1
1.2	Ziele der Arbeit . . . . .	4
1.3	Stand der Technik . . . . .	5
1.3.1	Hardware-Entwurf auf abstrakten Ebenen . . . . .	5
1.3.2	Gemeinsamer Entwurf von Hardware/Software . . . . .	6
1.3.3	Integration von CASE- und EDA-Werkzeugen und Methoden . . . . .	7
1.4	Gliederung . . . . .	10
<b>2</b>	<b>Hardware-Entwurf auf verschiedenen Abstraktionsebenen</b>	<b>11</b>
2.1	Register-Transfer-Ebene . . . . .	11
2.2	Algorithmische Ebene . . . . .	14
2.3	Systemebene . . . . .	16
<b>3</b>	<b>Kooperierende Prozesse und Objektorientierung</b>	<b>23</b>
3.1	Kooperierende Prozesse . . . . .	23
3.1.1	Blockierende / nicht blockierende Kommunikation . . . . .	24
3.1.2	Synchrone / asynchrone Kommunikation . . . . .	25
3.1.3	Nachrichtenbasierte Kommunikation und Kommunikation über gemeinsame Objekte . . . . .	26
3.1.4	Kooperierende Prozesse im Hardware-Entwurf . . . . .	28

3.2	Objektorientierung . . . . .	29
3.2.1	Anwendung der Objektorientierung im Hardware-Entwurf . . . . .	31
3.2.1.1	VHDL++ . . . . .	32
3.3	Prozeßbasierte versus objektorientierte Beschreibung . . . . .	35
3.3.1	Ähnlichkeiten . . . . .	35
3.3.2	Unterschiede . . . . .	35
3.4	Vorteile im Zusammenhang mit dem Hardware-Entwurf . . . . .	36
3.5	Unterschiede Software/Hardware . . . . .	37
4	<b>Ein SDL-basierter Ansatz zum abstrakten Systementwurf</b> . . . . .	<b>39</b>
4.1	SDL . . . . .	40
4.1.1	Blockdiagramme . . . . .	41
4.1.2	Prozeßdiagramme . . . . .	43
4.1.3	Datentypen . . . . .	44
4.1.4	Zeitmodell . . . . .	46
4.1.5	OSDL . . . . .	47
4.2	Hardware-Entwurf mit SDL auf unterschiedlichen Abstraktionsebenen . . . . .	49
4.2.1	Register-Transfer-Ebene . . . . .	50
4.2.1.1	Takt oder Setzen/Rücksetzen als Signale . . . . .	50
4.2.1.2	Signale im Sinne von Ein- und Ausgängen . . . . .	51
4.2.1.3	Einschränkung auf „Continuous“-Signale . . . . .	52
4.2.1.3.1	Impliziter Takt . . . . .	54
4.2.1.3.2	Expliziter Takt / Asynchrone Beschreibung . . . . .	54
4.2.1.4	Vorteile von SDL . . . . .	58
4.2.1.5	Probleme von SDL . . . . .	61
4.2.2	Algorithmische Ebene . . . . .	61

4.2.2.1	Vor- und Nachteile von SDL . . . . .	64
4.2.3	Systemebene . . . . .	64
4.2.3.1	Vorteile von SDL . . . . .	67
4.2.3.2	Probleme von SDL . . . . .	68
<b>5</b>	<b>Ein SDL-basiertes Entwurfssystem</b>	<b>71</b>
5.1	Konzept für ein SDL-basiertes Entwurfssystem . . . . .	71
5.2	Realisierung des SDL-basierten Entwurfssystems . . . . .	73
5.2.1	Überblick . . . . .	73
5.2.2	SDL-Sprachumfang . . . . .	75
5.2.2.1	Unterstützter SDL-Sprachumfang . . . . .	75
5.2.2.2	Einschränkungen im SDL-Sprachumfang . . . . .	77
5.2.2.3	Hardware-spezifische Erweiterungen . . . . .	77
5.2.3	Kanal- und Protokollbeschreibung . . . . .	80
5.2.3.1	Beschreibung der Kanalanschlüsse . . . . .	81
5.2.3.2	Beschreibung der Protokolle . . . . .	83
5.2.3.2.1	Anwendungsneutrale Protokolle . . . . .	85
5.2.3.2.2	Anwendungsspezifische Protokolle . . . . .	88
5.2.3.2.3	Problem der Protokollbeschreibung . . . . .	90
5.2.3.2.4	Alternativen . . . . .	93
5.2.3.3	Beschreibung der Kanalrealisierung . . . . .	94
5.2.4	Implementierungsbeschreibung . . . . .	94
5.2.4.1	Bibliotheken . . . . .	95
5.2.4.2	Teilsystem . . . . .	97
5.2.4.3	Komponentendeklaration . . . . .	98
5.2.4.4	Instanziierung . . . . .	98

5.2.4.4.1	Kanäle . . . . .	98
5.2.4.4.2	Prozesse . . . . .	99
5.2.5	SDL-nach-VHDL-Synthese . . . . .	101
5.2.5.1	Zielarchitektur . . . . .	103
5.2.5.2	Vorverarbeitung der Protokollbeschreibungen . . . . .	105
5.2.5.3	Synthese eines SDL-Prozesses . . . . .	106
5.2.5.3.1	„Entity“-Deklaration (Schnittstelle) . . . . .	107
5.2.5.3.2	„Architecture“-Deklaration (Schnittstelle) . . . . .	108
5.2.5.3.3	Empfangsteil . . . . .	108
5.2.5.3.4	Warteschlange . . . . .	108
5.2.5.3.5	Verarbeitungsteil . . . . .	110
5.2.5.3.6	Erzeugung der Netzliste für das Teilsystem . . . . .	115
5.2.5.3.7	Nachbearbeitung für Register-Transfer-Synthesesysteme . . . . .	115
<b>6</b>	<b>Ein Beispiel</b>	<b>121</b>
<b>7</b>	<b>Zusammenfassung und Ausblick</b>	<b>131</b>
7.1	Zusammenfassung . . . . .	131
7.2	Ausblick . . . . .	133
7.2.1	Hardware/Software-Co-Design . . . . .	133
7.2.2	Optimierungen und Verbesserungen . . . . .	135
	<b>Literaturverzeichnis</b>	<b>139</b>
<b>A</b>	<b>Syntax der Kanal- und Protokollbeschreibung</b>	<b>149</b>
<b>B</b>	<b>Syntax der Implementierungsbeschreibung</b>	<b>153</b>

# Verzeichnis der Bilder

1.1	Grobablauf einer Systementwicklung . . . . .	2
1.2	Der „One Year ASIC“ nach [Agne92] . . . . .	3
1.3	Grad der Automatisierung des Entwurfs auf verschiedenen Abstraktionsebenen . . . . .	5
2.1	Beispiel für eine algorithmische Beschreibung mit VHDL (aus [MiLD92]).	15
2.2	Abhängigkeit des Beschreibungsaufwandes vom Beschreibungsmittel und dem Problembereich . . . . .	18
2.3	Ansätze für Beschreibungsformen auf der Systemebene . . . . .	19
3.1	Klassendefinition für ein abstraktes RAM . . . . .	32
3.2	Klassendefinition für ein RAM mit seriellem Zugriff . . . . .	33
3.3	Klassendefinition für ein RAM mit parallelem Zugriff . . . . .	34
3.4	Verwendung der RAM Klasse . . . . .	34
4.1	SDL-Blockdiagramme . . . . .	42
4.2	Grundelemente eines SDL-Prozßdiagramms . . . . .	45
4.3	Prozßtypen und Vererbung in OSDL . . . . .	48
4.4	Blocktypen und Vererbung in OSDL . . . . .	49
4.5	Takt, Setzen/Rücksetzen als SDL-Signale . . . . .	50
4.6	SDL-Modell eines Multiplexers . . . . .	52
4.7	VHDL-Modell eines Multiplexers nach [Pulk93] . . . . .	52
4.8	Verwendung von SDL auf Register-Transfer-Ebene . . . . .	54

4.9	VHDL-Code für SDL-Beschreibung mit implizitem Takt . . . . .	55
4.10	VHDL-Code für asynchrones Verhalten und explizitem Takt (wait-until-Form) . . . . .	56
4.11	VHDL-Code für expliziten Takt (IF-Form) . . . . .	57
4.12	Beispiel mit gleichen Bedingungen in allen Zuständen . . . . .	57
4.13	Beispiel für die Register-Transfer-Ebene . . . . .	59
4.14	VHDL-Code für Beispiel 4.13 . . . . .	60
4.15	SDL-Beispiel auf der algorithmische Ebene . . . . .	63
4.16	VHDL-Code für die algorithmische Ebene . . . . .	63
4.17	SDL-Spezifikation eines Kellerspeichers . . . . .	65
4.18	Kaskadierbarer Kellerspeicher . . . . .	66
4.19	Nebenläufigkeit innerhalb eines SDL-Prozesses . . . . .	69
5.1	Der Entwurfsablauf . . . . .	72
5.2	Annotation von Ausführungszeiten . . . . .	73
5.3	Das SDL-nach-VHDL-Synthesystem . . . . .	74
5.4	Ausschnitt aus einem synthetisierbaren SDL-Prozeßdiagramm . . . . .	79
5.5	Das Kanalkonzept . . . . .	80
5.6	Aufbau einer Kanalbeschreibung . . . . .	82
5.7	Beispiel einer Schnittstellenbeschreibung . . . . .	83
5.8	Beschreibung der anwendungsneutralen Protokolle eines Handshake-Kanals (Schnittstelle siehe Bild 5.7) . . . . .	87
5.9	Anwendungsspezifische Protokolle auf tieferen Ebenen . . . . .	90
5.10	Anwendungsspezifische Protokolle auf der höchsten Ebene . . . . .	91
5.11	Beispiel für den internen Aufbau eines Kanals . . . . .	94
5.12	Ausschnitt aus einer Implementierungsbeschreibung . . . . .	96

5.13	Grobablauf der SDL-nach-VHDL-Synthese . . . . .	102
5.14	Zielarchitektur der SDL-nach-VHDL-Synthese . . . . .	103
5.15	Beispiel einer zyklischen Kommunikationsbeziehung . . . . .	105
5.16	Abbildung der Protokollschichten auf Prozedurschachtelung . . . . .	107
5.17	VHDL-Beschreibung der Schnittstelle, der Architektur und eines Empfangsteiles . . . . .	109
5.18	VHDL-Beschreibung für die Warteschlange . . . . .	111
5.19	VHDL-Beschreibung für den Verarbeitungsteil (I) . . . . .	113
5.20	VHDL-Beschreibung für den Verarbeitungsteil (II) . . . . .	114
5.21	Beispiel zur Demonstration der Nachbearbeitungsschritte . . . . .	117
5.22	Beispiel nach der Expansion der Prozedure receive . . . . .	118
5.23	Beispiel nach der Wait-Faltung . . . . .	119
6.1	Blockdiagramm des Systems . . . . .	121
6.2	Prozeßdiagramm des Prozessors . . . . .	122
6.3	Protokollbeschreibung des Kanals zum Speicher . . . . .	124
6.4	Implementierungsbeschreibung für das System . . . . .	125
6.5	Ausschnitt aus einer Protokollbeschreibung für einen Speicher mit seri- ellem Zugriff . . . . .	126
6.6	Generierter VHDL-Code für den Prozessor (Teil 1) . . . . .	128
6.7	Generierter VHDL-Code für den Prozessor (Teil 2) . . . . .	129
6.8	Synthetisierte Netzliste des Prozessors . . . . .	130
7.1	Kommunikation zwischen Hardware und Software . . . . .	135
7.2	Mögliche Realisierung einer Zustandshierarchie in SDL . . . . .	137