

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation und Problemstellung	3
1.2	Aufbau der Arbeit	5
2	Grundlagen	7
2.1	Entwurfsmethoden für eingebettete Systeme	7
2.1.1	Allgemeine Entwurfsmethoden für Software	7
2.1.2	Entwurfsmethoden für Hardware/Software-Systeme	8
2.1.3	Plattformbasierter Entwurf	10
2.1.4	Prototyping	12
2.2	Hardware-Entwicklung für eingebettete Systeme	14
2.2.1	Hardware-Beschreibungssprachen	14
2.2.2	Hardware-Synthese	15
2.2.3	Verifikation der Hardware	17
2.3	Software-Entwicklung für eingebettete Systeme	18
2.3.1	Entwicklungsumgebungen	18
2.3.2	Debugging der Software	19
2.4	Hardware-Emulation	20
2.4.1	Überblick über FPGAs	21
2.4.2	Funktionsweise SRAM-basierter FPGAs	21
2.4.3	Hardware-Architektur von FPGAs	23
2.4.4	Debugging mit integrierten Logikanalysatoren	28
3	Stand der Technik	31
3.1	Software-Entwicklung in frühen Entwurfsphasen	31
3.1.1	Entwicklung von Software-Algorithmen mit Befehlssatzsimulatoren	32
3.1.2	Einsatz von Java in eingebetteten Systemen	32
3.1.3	Transaktionsbasierte Modellierung auf hohen Abstraktionsebenen	33
3.1.4	Software-Entwicklung mit Evaluierungsboards	34
3.2	Entwicklung eingebetteter Hardware/Software-Systeme	35

3.2.1	Simulation von Hardware-Komponenten	36
3.2.2	Kombination von BSS mit HDL-Simulator	38
3.2.3	Emulation von Hardware-Komponenten	39
3.2.4	Kombination von BSS mit Hardware-Emulator	41
3.2.5	Kombination aus Prozessor-ASIC und FPGA	42
3.2.6	Integration von Prozessoren in FPGAs	43
3.2.7	Formale Verifikation	44
4	Bewertung des Stands der Technik	45
4.1	Bewertung der Entwurfsmethoden für die Software-Entwicklung . .	45
4.2	Bewertung der Entwurfsmethoden für die Entwicklung komplexer Hardware-Komponenten	46
4.3	Zielsetzung einer neuen Entwurfsmethodik für eingebettete Hardware/- Software-Systeme	48
5	Konzept einer Entwicklungsumgebung für den Entwurf eingebet- teter Hardware/Software-Systeme	53
5.1	Grundkonzept der Entwurfsmethodik	53
5.2	Beschleunigung der Anwendungsentwicklung durch Emulation . . .	55
5.2.1	Prototyping von ASIC-IP-Kernen auf FPGAs	55
5.2.2	Methodik zur Lösung des Ressourcenproblems	59
5.2.3	Architektur der rekonfigurierbaren Entwicklungsumgebung	60
5.2.4	Integration eines System-on-Chip auf dem System	63
5.2.5	Hardware-nahe Software-Entwicklung unter harten Ressour- ceneinschränkungen	67
5.3	Parallele Entwicklung von Hardware-Komponenten und hardware- naher Software	69
5.3.1	Überblick	70
5.3.2	Simulation des eingebetteten Hardware/Software-Systems .	72
5.3.3	Entwicklung von Hardware-Komponenten auf einem Board	74
5.3.4	Hardware-Entwicklung mit Hilfe von Erweiterungsplattformen	76
5.3.5	Emulation und Hardware-Debugging des SoC	78
5.3.6	Software-Entwicklung mit einer Cross-Entwicklungsumge- bung	80
5.4	Zusammenfassung	82
6	Hardware/Software-Entwicklung mit einem emulationsbasierten Mikrocontroller-IP-Kern	85
6.1	TriCore1-Mikrocontroller	85

6.1.1	Überblick	86
6.1.2	Architektur	86
6.1.3	Anbindung von Peripheriemodulen	88
6.1.4	Programmiermodell und Unterbrechungsbehandlung	89
6.1.5	TriCore1-Plattformen	90
6.2	Rapid-Prototyping-System Spyder	91
6.3	Emulation des TriCore1-Mikrokontroller-IP-Kerns	93
6.3.1	Technologische Anpassung des ASIC-IP-Kerns	93
6.3.2	Realisierung komplexer SoC-Prototypen unter harten Ressourcenbeschränkungen	96
6.3.3	Zwischenergebnis	104
6.3.4	Integration des TriCore1-Mikrokontroller-IP-Kerns auf dem Spyder-System	107
6.3.5	Zwischenergebnis	113
6.4	Entwicklung eingebetteter Hardware/Software-Systeme	115
6.4.1	Beispiel eines typischen SoC-Bussystems	115
6.4.2	Entwicklung und Integration neuer Hardware-Komponenten	117
6.4.3	Plattformbasierter Entwurf mit Erweiterungsplattformen	120
6.4.4	Verifikation der Hardware-Komponenten	128
6.4.5	Zwischenergebnis	128
6.5	Zusammenfassung	129
7	Ergebnisse	131
7.1	Entwicklung eingebetteter Software auf Hardware-Modellen	131
7.1.1	Implementierung einer TC1MP-S-Plattform	132
7.1.2	Laufzeitmessungen für eingebettete Software	137
7.1.3	Architekturgenaue Emulation des TriCore1-Mikrokontroller-Kerns	144
7.1.4	Zusammenfassung	147
7.2	Entwicklung eingebetteter Hardware/Software-Systeme	147
7.2.1	Überblick über die Hardware-Komponenten	148
7.2.2	Syntheseergebnisse	149
7.2.3	Debugging von Hardware und Software	153
7.2.4	Laufzeitmessungen	156
7.2.5	Zusammenfassung	159
8	Zusammenfassung	161
A	Abkürzungen	163