

Inhaltsverzeichnis

1	Einführung	1
1.1	Motivation	1
1.1.1	Entwicklungsfluß	3
1.1.2	Kritische Zeitpfade im Entwicklungsfluß	4
1.2	Virtueller Test	6
1.3	Ergebnisse und Gliederung der Arbeit	8
1.3.1	Ergebnisse	8
1.3.2	Gliederung	8
2	Grundlagen zum virtuellen Test	11
2.1	Analyse der Hardware eines Mixed-Signal-Testers	11
2.1.1	Strukturelle Gliederung eines Testsystems	11
2.1.2	Bussysteme	13
2.1.3	Spannungsversorgung	17
2.1.4	Instrumente	17
2.1.5	Unterschiede in der Steuerungs-Architektur	24
2.1.6	Hierarchische Aufteilung der Testsystemressourcen	26

2.1.7	Trends	27
2.1.8	Zusammenfassung der Hardware-Analyse	29
2.2	Analyse der Software-Architektur	30
2.2.1	Bedienungsumgebung	30
2.2.2	Hierarchischer Aufbau der Programmierungsumgebung	31
2.2.3	Aufbau eines Testprogramms	32
2.2.4	Programmierung der Pinelektronik (DPIN)	33
2.2.5	Zusammenfassung der Software-Analyse	37
3	Stand der Technik	39
3.1	Vergleich kommerzieller Virtual Test Produkte	39
3.1.1	Untersuchungskriterien	39
3.1.2	Produkte	41
3.1.3	Produktvergleich / Diskussion	51
3.2	Publizierte Ansätze zur Modellbildung	54
3.2.1	Noise und Jitter im virtuellen Test [HEL96-1] [HEL96-2]	54
3.2.2	Testsystemsimulation mit hSpice	57
3.2.3	Wiederverwendbarkeit von DUT-Systemmodellen	58
3.2.4	Benutzerdefinierte Virtual Test Umgebung	59
3.3	Kopplungsmechanismen	61
3.3.1	Emulator-Docking	61
3.3.2	Testprogramm-Docking	61
3.4	Zusammenfassung der Analyse und resultierende Ziele	61

4	Implementierung eines virtuellen Testsystems	65
4.1	Vorhandene Komponenten	65
4.1.1	SPACE	65
4.1.2	Simulator	67
4.1.3	Interface SPACE-Simulator	68
4.2	Anforderung, Konzept und Realisierung des virtuellen Tests	70
4.2.1	Bedienungsumgebung	71
4.2.2	Konfiguration der Simulation	71
4.2.3	Simulationssteuerung	73
4.2.4	Datenverwaltung	85
4.2.5	Modellierungstechnik und Geschwindigkeit	89
4.2.6	Ergebnisanalyse	95
4.3	Zusammenfassung der Implementierung	98
5	Modellierungskonzept für die Entwicklung einer neuen Testsystemarchitektur	99
5.1	Anforderungen	99
5.2	Konzept zur Realisierung der Modellierung neuer Testsystemarchitekturen	100
5.2.1	Hierarchie, Klassen und Versionen bei der Modellierung	100
5.2.2	Datenbank	101
5.2.3	Ressourcen-Editor	101
5.2.4	Beobachter	103
5.2.5	Schematic-Entry-Tool	104
5.2.6	Betriebssystem	105
5.2.7	Struktur der Entwicklungsumgebung	105
5.3	Zusammenfassung	106

6	Verifikation der Modelle	107
6.1	Vorhandene Testprogramme	107
6.2	Ausgewählte Testschritte des Audio Codecs	108
6.2.1	Kontakttest	109
6.2.2	Test der Ausgangstreiber	109
6.2.3	Leckstromtest	111
6.2.4	CD-to-Line-Test	111
6.2.5	DAC-to-Line-Test	112
6.2.6	ICC-Test	113
6.3	Einschwingvorgänge Simulation und Realität	113
6.3.1	Meßergebnisse	114
6.3.2	Simulationsergebnisse	118
6.4	Evaluiierung	123
6.4.1	Evaluiierungswoche	123
6.4.2	β -Test	124
6.5	Auswertung	125
7	Zusammenfassung und Ausblick	127
7.1	Zusammenfassung	127
7.2	Ausblick	128

A	Software-Architektur	129
A.1	Software-Hierarchie	129
A.1.1	Basic-Instrumente	129
A.1.2	Master-Instrumente	129
A.1.3	Super-Instrumente	132
A.2	Testprogramm-Aufbau	133
A.2.1	Initialisierung	133
A.2.2	Setup	134
A.2.3	Testschritte	134
A.2.4	Shutdown	136
A.3	Pattern-Dateiformate	136
A.3.1	WDB	136
A.3.2	WGL-File	139
A.3.3	PPS-Format	141
B	Implementierung	143
B.1	Datenfeld der Digitalen Pinelektronik - DPIN	143
B.1.1	PDCL und PMU	143
B.1.2	Patterndaten	145
B.2	Verhaltensmodellierung der aktiven Last	150
B.2.1	Steuerung der aktiven Last	152
B.2.2	Modellierung der aktiven Last	152
B.2.3	Referenzmodell der aktiven Last	154

B.3	Patternabarbeitung auf verschiedenen Abstraktionsebenen	156
B.3.1	Pattern-ID	156
B.3.2	Starre Eventliste	159
B.3.3	Sequencer	161
B.3.4	Master-Clock	162
C	Abkürzungen	163
D	Definitionen	165
	Literaturverzeichnis	171