



Volker Dörsing (Autor)
Adrian Knoth (Autor)
Wolfgang Koch (Autor)
David Neuhäuser (Autor)
Andreas Reinsch (Autor)
Ralf Seidler (Autor)
Daniel Walther (Autor)
Martin Bücker (Herausgeber)

Experimente der Technischen Informatik Praxisorientierte Trainingseinheiten

Hrsg. H. Martin Bücker

Experimente zur Technischen Informatik Praxisorientierte Trainingseinheiten

Volker Dörsing
Adrian Knoth
Wolfgang Koch
David Neuhäuser

Andreas Reinsch
Ralf Seidler
Daniel Walther



Cuvillier Verlag Göttingen
Internationaler wissenschaftlicher Fachverlag

<https://cuvillier.de/de/shop/publications/8928>

Copyright:

Cuvillier Verlag, Inhaberin Annette Jentsch-Cuvillier, Nonnenstieg 8, 37075 Göttingen,
Germany

Telefon: +49 (0)551 54724-0, E-Mail: info@cuvillier.de, Website: <https://cuvillier.de>

Inhaltsverzeichnis

I. Reduced Instruction Set Computer (RISC)	1
1. Grundlagen P: DLXJ-RISC-Prozessor	3
1.1. Jenaer Version der DLX-Architektur	4
1.2. Programmiermodell des DLXJ-Prozessors	5
1.2.1. Befehlsklassen	5
1.2.2. Befehlsformate	7
1.3. Struktur des DLXJ-Prozessors	8
1.3.1. Prozessorkern	10
1.3.2. Speichereinheit	11
1.3.3. Display-Interface	11
1.3.4. Mikrocontroller-Interface	11
1.4. Interner Aufbau des Prozessorkerns	12
1.4.1. Datenpfad	12
1.4.2. Steuerung	15
1.4.3. Grundschnitte der Befehlsausführung	17
1.4.4. Schaltwerk	20
1.5. Entwicklungsumgebung	23
1.5.1. Assemblierung von Programmen	24
1.5.2. Analyse und Synthese	24
1.5.3. Simulation	25
1.5.4. DLXJ-Debugger	25
Anhang 1.A. VHDL-Beschreibungen	28
1.A.1. Arithmetic Logic Unit	28
1.A.2. Decoder 1	29
1.A.3. Decoder 2	30
1.A.4. Decoder 3	31
1.A.5. Steuerkonstanten	32
1.A.6. Zustandsüberföhrungsfunktion	36
1.A.7. Zustandspeicher	38
1.A.8. Ergebnisfunktion	39
Anhang 1.B. Programmbeispiel	42
Literatur	43
2. Versuch P1: Computer-Aided Engineering (CAE-)Werkzeuge	45
2.1. Überblick über die Versuche zum DLXJ-RISC-Prozessor	46
2.1.1. Computer Aided Engineering	46
2.1.2. FPGA Xilinx Spartan-3AN	47
2.2. Entwicklungsprozess	47
2.2.1. Simulationsmodell	47
2.2.2. Entwurfsbeispiel D-Latch	48

2.2.3.	Funktionale Simulation	51
2.2.4.	Synthese	51
2.2.5.	Implementierung	52
2.2.6.	Zeitbehaftete Simulation	52
2.3.	Versuchsvorbereitung	53
2.4.	Versuchsdurchführung	55
2.4.1.	Funktionale Simulation	55
2.4.2.	Synthese und Implementierung	56
2.4.3.	Zeitbehaftete Simulation	58
2.4.4.	Erprobung	58
2.5.	Aufgabenstellung	59
2.5.1.	Funktionale Simulation	59
2.5.2.	Synthese und Implementierung	60
2.5.3.	Zeitbehaftete Simulation	60
2.5.4.	Erprobung der Experimentalschaltung	61
2.6.	Bemerkungen zu weiterführender Literatur	61
	Literatur	61
3.	Versuch P2: Befehlszähler eines RISC-Prozessors	63
3.1.	Befehlszähler im Datenpfad des Prozessorkerns	64
3.2.	Funktionsweise des Befehlszählers	64
3.2.1.	Aufbau des Befehlszählers	64
3.2.2.	Berechnung einer Befehlsadresse	66
3.3.	Versuchsvorbereitung	67
3.3.1.	DLXJ-Prozessor	67
3.3.2.	Maßnahmen am Arbeitsplatz	67
3.4.	Versuchsdurchführung	68
3.4.1.	Simulation, Synthese und Implementierung	68
3.4.2.	Erprobung	68
3.5.	Aufgabenstellung	70
3.5.1.	Funktionale Simulation	70
3.5.2.	Zeitbehaftete Simulation	70
3.5.3.	Erprobung der Experimentalschaltung	70
3.6.	Bemerkungen zu weiterführender Literatur	71
	Literatur	71
4.	Versuch P3: Aufbau und Funktionsweise eines RISC-Prozessors	73
4.1.	RISC-Prozessoren	74
4.2.	Versuchsvorbereitung	74
4.2.1.	DLXJ-Prozessor	74
4.2.2.	Maßnahmen am Arbeitsplatz	75
4.3.	Versuchsdurchführung und Aufgabenstellung	76
4.3.1.	Implementierung zusätzlicher Befehle	76
4.3.2.	Test der zusätzlichen Befehle und des Assemblerprogrammes	77

4.4. Versuchsauswertung	78
4.5. Bemerkungen zu weiterführender Literatur	78
Anhang 4.A. Kodierung des erweiterten DLXJ-Befehlssatzes	78
Anhang 4.B. Testprogramm für die Additionsbefehle	79
Literatur	79
5. Versuch P4: Assemblerprogrammierung eines RISC-Prozessors	81
5.1. Assemblerprogrammierung und Debugging	82
5.2. Versuchsvorbereitung	82
5.2.1. Debugging des DLXJ-Prozessors	82
5.2.2. Fibonacci-Zahlen	82
5.2.3. Codierung	83
5.2.4. Maßnahmen am Arbeitsplatz	83
5.3. Versuchsdurchführung und Aufgabenstellung	83
5.3.1. Implementierung und Test der Assemblerprogramme	83
5.4. Versuchsauswertung	85
5.5. Bemerkungen zu weiterführender Literatur	85
Anhang 5.A. Dual-BCD-Umsetzung	85
Anhang 5.B. Programm für die Dual-BCD-Umsetzung	86
Literatur	88
II. Graphics Processing Unit (GPU)	89
6. Versuch G: GPU Programmierung	91
6.1. Massiv-parallele Programmierung von Grafikkarten	92
6.2. Versuchsvorbereitung	93
6.2.1. Begriffsbildung	93
6.2.2. Grundlagen der Bildverarbeitung	93
6.2.3. Grundlagen des GPU Computing	97
6.3. Versuchsdurchführung und Aufgabenstellung G1	104
6.3.1. Arbeitsumgebung	104
6.3.2. Helligkeit/Kontrast anpassen	105
6.3.3. Bild spiegeln	105
6.3.4. Graubild erstellen	105
6.3.5. Kantendetektion mit Sobel-Filter	106
6.3.6. Debugging	106
6.4. Versuchsdurchführung und Aufgabenstellung G2	107
6.4.1. Performance-Analyse	107
6.4.2. Performance-Steigerung	108
6.5. Bemerkungen zu weiterführender Literatur	110
Literatur	111

III. Single Instruction, Multiple Data (SIMD)	113
7. Versuch S: Programmierung mit Streaming SIMD Extensions (SSE)	115
7.1. Parallelisierung auf der Ebene von Befehlen	116
7.2. Versuchsvorbereitung	117
7.2.1. Begriffsbildung	117
7.2.2. Grundlagen der Bildverarbeitung	117
7.2.3. Verwendetes Speicherlayout	118
7.3. Funktionsreferenz	120
7.3.1. SSE-Schnellübersicht	120
7.3.2. Weiterführende Informationen zu SSE	120
7.3.3. Hilfsfunktionen	121
7.3.4. Variablen und Datentypen	121
7.4. Versuchsdurchführung und Aufgabenstellung	122
7.4.1. Anzeige eines schwarzen Bildes für zwei Sekunden	122
7.4.2. Einblenden des Videos <i>A</i> für zwei Sekunden	122
7.4.3. Anzeige des Videos <i>A</i> für zwei Sekunden	122
7.4.4. Überblendung des Videos <i>A</i> in Video <i>B</i> über vier Sekunden	122
7.4.5. Anzeige des Videos <i>B</i> für zwei Sekunden	123
7.4.6. Ausblendung des Videos <i>B</i> für zwei Sekunden nach Grün	123
7.4.7. Anzeige eines grünen Bildes für eine Sekunde	123
7.4.8. Assembler-Output	123
7.5. Bemerkungen zu weiterführender Literatur	123
Literatur	124
IV. Digitalelektronik	125
8. Versuch D: Digitalelektronik	127
8.1. Versuchsvorbereitung	128
8.1.1. Grundlagen von MOS-Feldeffekttransistoren	128
8.1.2. Analyse einer vorgegebenen Beispielschaltung	129
8.1.3. Spezielle Aspekte der Digitalelektronik	130
8.1.4. Hinweise zur Mess-Hardware	130
8.2. Kurzanleitung PSpice	130
8.2.1. Spice-Programme	131
8.2.2. Gebrauchsanweisung PSpice	132
8.2.3. Modelle für MOSFETs	133
8.3. Versuchsdurchführung und Aufgabenstellung	134
8.3.1. Ausgangskennlinie simulieren (A1)	134
8.3.2. Übertragungskennlinie simulieren (A2)	134
8.3.3. Übertragungskennlinie messen (A3)	135
8.3.4. Arbeitsgerade einzeichnen (V1)	135
8.3.5. Ein- und Ausschaltverhalten simulieren (V2)	135

8.3.6. Übertragungskennlinie CMOS-Inverter messen (A4)	135
8.3.7. Strom im CMOS-Inverter simulieren (A5)	136
8.3.8. NAND-Gatter oder NOR-Gatter aufbauen (A6)	136
8.3.9. FF-Zelle aufbauen (V3)	136
8.4. Bemerkungen zu weiterführender Literatur	136
Literatur	136
V. Netzwerke	139
9. Versuch N: Rechner-Netzwerke	141
9.1. Digitale Kommunikation zwischen Rechnern	142
9.2. Versuchsvorbereitung	142
9.3. Versuchsdurchführung und Aufgabenstellung	143
9.3.1. Grundlegende (lokale) Konfiguration	143
9.3.2. Kommunikation zwischen den Versuchsrechnern	144
9.3.3. Routing	145
9.3.4. Routenverfolgung	145
9.3.5. DNS abhören	146
9.3.6. URL-Extraktion aus Netzwerk-Dumps	147
9.3.7. Upload der Protokolldateien	147
9.4. Bemerkungen zu weiterführender Literatur	149
Literatur	149
VI. Rechnerarithmetik	151
10. Versuch R: Digitale Rechenschaltungen	153
10.1. Versuchsvorbereitung	154
10.2. Versuchsdurchführung	154
10.2.1. Versuchsprotokoll	154
10.2.2. Hinweise zum Xilinx [®] Integrated Software Environment	155
10.2.3. Hinweise zu Skripten [®] und Synopsys [®]	159
10.3. Aufgabenstellung	160
10.3.1. Volladdierer	160
10.3.2. Ripple-Carry-Addierer	161
10.3.3. Carry-Skip-Addierer	162
10.3.4. Addierer/Subtrahierer	162
10.4. Bemerkungen zu weiterführender Literatur	165
Literatur	165