

CARL HANSER VERLAG

Christian Siemers

Prozessorbau

Eine konstruktive Einführung in das Hardware/Software-Interface

3-446-19330-8

www.hanser.de

Vorwort

50 Jahre nach der Einführung von Prozessoren und 25 Jahre nach dem ersten Mikroprozessor ist es nun möglich, eigene ICs zu designen und fertigzustellen, ohne eine Fabrik im Hintergrund zu haben. Eine völlig andere Klasse von programmierbaren Systemen, die feldprogrammierbaren Bausteine, sind zwischenzeitlich in einer Größe erhältlich, die für die Integration kleinerer Mikroprozessoren geeignet ist.

Damit ist etwas möglich geworden, das einige Jahrzehnte undurchführbar erschien. Jeder Entwickler kann seinen eigenen Prozessor designen und herstellen. Folglich ist eine Anpassung an die jeweilige Applikation möglich, und dies kann mit allen Konsequenzen erfolgen.

Die neuen Freiheitsgrade, die hierdurch erhalten werden, lassen sich für diverse Optimierungen nutzen: Schnellere Applikationen ohne Takterhöhung, Echtzeitfähigkeiten mit sehr geringem Jitter der Reaktionszeit in der Größenordnung eines Prozessortakts, dies ohne zusätzliche Hardware, bis hin zu neuen, experimentellen Prozessorarchitekturen werden hierdurch möglich.

Dieses Buch wendet sich an diejenigen, die sich diesen Herausforderungen und Möglichkeiten stellen wollen. Da es jedoch nicht die Phantasie aller Entwickler zur Variierung der bisherigen Prozessorarchitekturen vorwegnehmen kann, wird in den Kapiteln im Buch selbst die Architektur von Prozessoren im Detail behandelt. Ziel dieser Verfahrensweise ist eine konstruktive Einführung in dieses Thema. Zu diesem Zweck werden zwei Prozessormodelle vollständig konzipiert und als VHDL-Modelle erstellt.

Die genaue Kenntnis nicht nur der Prozessorarchitektur, sondern auch der Wechselwirkung zwischen Hard- und Software, ist für alle Ansätze, eine für die spezielle Applikation optimierten Prozessor zu bauen, notwendig. Der Nebeneffekt dieser Darstellungen in den Kapiteln 2 bis 5 besteht natürlich darin, daß dieses Buch auch zur Lehre und zum Verständnis der Prozessorarchitektur geeignet ist.

Die Kapitel 6 und 7 befassen sich schließlich mit der Variation der dargestellten Modelle. Kapitel 6 nutzt hierzu den klassischen Ansatz, die Bearbeitung zu beschleunigen, indem eine für Phasenpipelining geeignete RISC-Architektur in den Grundzügen eingeführt wird.

Kapitel 7 schließlich geht auf Möglichkeiten zu eigenständigen Änderung von vorhandenen Prozessordesigns ein. Der Leser sollte gerade dieses Kapitel als Anregung zu eigenen Versuchen, die klassischen Strukturen aufzubrechen und die Grenze zwischen Hard- und Software variabel zu gestalten, auffassen.

Zur praktischen Durchführung sind einige Programme als Voll- oder Sharewareversionen beigelegt. Leider existiert auf dem Gebiet VHDL noch keine freie Software zur Simulation aller Eingaben, so daß auch mit den beigelegten Programmen nur eingeschränkte Möglichkeiten zum Test gegeben sind.

Der Autor möchte auf diesem Weg den beteiligten Personen und Firmen für ihre Bereitschaft danken, diese Programme beizusteuern. Namentlich sind es Hr. Dr. Hans-Peter Hohe, Heiligenstadt, Hr. Ralf Kimmelmann, Fa. INt, Martinsried sowie

die Herren Wolfgang Reis und Jörg Siemers, Fa. Vantis, München, die mit tatkräftiger Hilfe auf dem Sektor der Software zur Vervollständigung des Buches beigetragen haben.

Der besondere Dank des Autors gilt zwei Damen, die sich um das Buch selbst und das Innenleben des Verfassers gekümmert haben. Hier ist in erster Linie die Lektorin des Fachbuchverlages Leipzig, Frau Erika Hotho, zu erwähnen. Ihr gilt der Dank, den Autor so sanft gedrängt zu haben, daß er sich nicht bedrängt fühlte und trotzdem dieses Buch vollendet hat. Zudem hat sie sich immer wieder die Zeit genommen, das noch unvollendete Werk Korrektur zu lesen.

Meiner Frau, die zweite entscheidende Dame für dieses Buch, möchte ich ganz besonders für ihre Doppelrolle danken. Einerseits mußte sie immer wieder Mut zusprechen, das Werk weiterzuführen, um dabei gleichzeitig zu ertragen, daß auch oder gerade Wochenenden hierfür notwendig waren, und andererseits durfte sie auch noch Korrektur lesen, eine Aufgabe, die sie mit Sachverstand und Geschick gemeistert hat. Ohne sie wäre dieses Buch nicht fertig geworden.

Hildesheim, im Dezember 1998

Inhaltsverzeichnis

1	Komponenten zum Prozessorbau	11
1.1	Programmierbare Logik.....	12
1.1.1	Komponenten in PLDs.....	12
1.1.1.1	Disjunktive Normalform als Schaltnetztyp innerhalb PLDs	13
1.1.1.2	Look-up-Table-basierte Schaltnetze	14
1.1.1.3	Multiplexerbasierte Schaltnetze.....	15
1.1.1.4	Speicherung von Schaltvariablen.....	15
1.1.2	Klasseneinteilung der PLDs.....	17
1.1.2.1	Simple PLDs	17
1.1.2.2	High Density PLDs (HDPLDs).....	18
1.2	Endliche Automaten	21
1.2.1	Einfache Automaten	22
1.2.2	Komplexe, kooperierende Automaten	23
1.3	Hardwarebeschreibungssprachen	25
1.3.1	Boolescher Assembler	25
1.3.2	Finite State Machine Syntax	26
1.3.3	Very High Speed Integrated Circuits Hardware Description Language (VHDL).....	28
1.3.3.1	Der Entwurf elektronischer Systeme	28
1.3.3.2	Der grundsätzliche Aufbau einer VHDL-Beschreibung.....	29
1.3.3.3	Beschreibungselemente in VHDL	31
1.4	Zur Darstellung in diesem Buch.....	31
2	Von-Neumann-Rechnermodell.....	33
2.1	Grundkonzept.....	33
2.2	Bussysteme	37
2.2.1	Grundsätzlicher Aufbau: in- und externe Busse.....	37
2.2.2	Adreß-, Daten- und Steuerbus im zeitlichen Verhalten.....	39
2.3	Speicher	43
2.4	Ein- und Ausgabelemente	43
2.5	Prozessorregister im Leitwerk	44
2.5.1	Programmzähler.....	45
2.5.2	Statusregister.....	46
2.5.3	Stackpointer	47
2.6	Rechenwerk	48
2.6.1	Datenregister	49
2.6.2	Datenpfade im Rechenwerk.....	49
2.6.3	Aufbau der ALU	50
2.7	Interpretation der binären Darstellung als Datenwerte.....	51
2.7.1	Darstellung binärwertiger Vektoren	51
2.7.2	Darstellung ganzer Zahlen	52
2.7.3	Darstellung von Gleitkommazahlen	53

2.7.4	Darstellung alphanumerischer Daten.....	54
2.8	Phasen der Befehlsbearbeitung.....	55
2.9	Adressierungsarten im Von-Neumann-Prozessor	58
2.10	Klassifizierung von Prozessoren gemäß Operandenzugriff.....	62
2.11	Interrupt-Konzept als Erweiterung des Von-Neumann-Prozessors	64
2.11.1	Konzept der Behandlung von Interrupts.....	64
2.11.2	Anwendungen	66
2.11.3	Notwendige Erweiterungen im Prozessor	67
3	Modellarchitektur MPM1	69
3.1	MPM1-Konzept	70
3.1.1	Speichermodell des MPM1.....	71
3.1.2	Festlegung der Prozessorklasse	72
3.1.3	Programmiermodell des MPM1	73
3.1.4	Codierung der Befehle und Adressierungsarten im MPM1	76
3.1.5	Definition der Phasen im Befehlsablauf.....	78
3.1.6	Verhaltensbeschreibung des MPM1	79
3.2	Phasen der Befehlsverarbeitung in der detaillierten Modellierung.....	81
3.2.1	Fetch-Phase	82
3.2.2	Decode-Phase.....	84
3.2.3	Load-Phase.....	85
3.2.4	Execute-Phase	88
3.2.4.1	Execute-Phase für Sprungbefehle	89
3.2.4.2	Execute-Phase für Nichtsprungbefehle.....	89
3.2.5	Write-Back-Phase	90
3.2.6	Steuerwerk mit Phasensteuerung als Konsequenz der Abläufe	92
3.2.7	Verhalten des Prozessors nach Reset.....	94
3.3	MPM1: Modell-CPU mit 4 Bit Verarbeitungsbreite.....	95
3.4	Testprogramme	98
4	Integration des Rechenwerks	100
4.1	Gesamtkonzept des Rechenwerks	100
4.2	Logische Basisverknüpfungen.....	102
4.2.1	Logikfunktionen.....	102
4.2.2	Schiebe- und Rotationsbefehle	103
4.3	Arithmetische Basisverknüpfungen.....	104
4.3.1	Addition zweier Binärzahlen	104
4.3.1.1	Der sequentielle Addierer	104
4.3.1.2	Ripple Carry Adder	105
4.3.1.3	Paralleler Addierer	106
4.3.1.4	Carry Look Ahead Adder (CLAA).....	107
4.3.2	Verallgemeinerte Addierer	109
4.3.3	Subtraktion.....	111
4.3.4	Vorzeichenbehaftete Addition/Subtraktion	113
4.3.5	Multiplikation	115

4.3.6	Division.....	119
4.3.7	Schiebe- und Rotationsoperationen (2).....	121
4.4	Zusammenfügung zur arithmetisch-logischen Einheit.....	122
4.4.1	Nutzung von Multiplexern zur Operationsauswahl.....	122
4.4.2	Eine 4-Bit-ALU mit Bestimmung der Flags.....	123
4.5	MPM1-Rechenwerk.....	126
5	Erweiterung zur Modellarchitektur MPM2.....	129
5.1	Feinspezifikation des MPM2.....	129
5.1.1	Speichermodell des MPM2.....	129
5.1.2	Programmiermodell des MPM2.....	131
5.1.3	Codierung der Befehle und Adressierungsarten im MPM2.....	137
5.1.4	Definition der Phasen im Befehlsablauf.....	140
5.1.5	Interrupt-Request-Konzept des MPM2.....	141
5.1.6	Eine Verhaltensbeschreibung des MPM2.....	143
5.2	Strukturelle Beschreibung der spezifischen Erweiterungen zum MPM2.....	145
5.2.1	Durchführung des Reset im MPM2.....	147
5.2.2	Realisierung des Interrupt-Request-Konzepts im MPM2.....	148
5.2.3	Unterprogrammaufrufe und -rücksprünge.....	152
5.2.4	Stackpointer.....	153
5.2.5	Adressierungsarten und Adreß-ALU.....	155
5.2.6	Akkumulator und Indexregister im MPM2.....	157
5.2.7	Statusregister im MPM2.....	159
5.2.8	Arithmetisch-logische Einheit im MPM2.....	160
5.2.9	Decode-Einheit im MPM2.....	161
5.2.10	Steuerwerk und Befehlsphasen im MPM2.....	162
5.3	MPM2-Struktur.....	165
5.4	Testprogramme für MPM2.....	167
6	Pipelining und RISC-Prozessoren.....	169
6.1	Basiskonzept Pipelining.....	170
6.1.1	Ziel des Pipelinings.....	171
6.1.2	Anwendung von Pipelining auf MPM2-Befehle.....	172
6.1.3	Realisierungsziele eines optimalen Phasenpipelinings.....	174
6.2	Pipelinehazards.....	175
6.2.1	Performance von Pipeline bei Einfügung von Wartezyklen.....	175
6.2.2	Strukturhazards.....	177
6.2.3	Kontrollflußhazards.....	178
6.2.4	Datenhazards.....	180
6.3	MPM3: Konsequenzen für ein Prozessormodell mit Phasenpipelining... ..	180
6.3.1	Prozessor-Architekturklasse und Programmiermodell.....	181
6.3.2	Instruktionssatz MPM3.....	181
6.3.2.1	Transferbefehle.....	182
6.3.2.2	Arithmetisch-logische Befehle im MPM3.....	184
6.3.2.3	Flagbefehle.....	185

6.3.2.4 Kontrollfluß-Befehle	185
6.3.3 Ablauf der Instruktionen im MPM3	186
6.4 Unterprogrammssprünge und Ausnahmebehandlung bei Pipelining	187
7 Prozessorbau und Hardware-Software-Co-Design.....	189
7.1 Hardware-Software-Co-Design	189
7.1.1 Applikationsdomänen	190
7.1.2 Co-Design von Embedded-Systemen	191
7.1.3 Co-Design und Instruction Set Architekturen	192
7.2 Ansätze für neue Prozessormodelle.....	193
7.2.1 Variationen im Befehlssatz.....	193
7.2.2 Ausnahmebehandlungen und Co-Design	196
7.3 Applikationsspezifische Prozessoren in der Zusammenfassung	199
Glossar.....	200
Literaturverzeichnis.....	203
Sachwortverzeichnis.....	204

Zur CD in diesem Buch

Die beigefügte CD enthält einige Softwarekomponenten zu den Kapiteln in diesem Buch, die sich in zwei Kategorien einteilen lassen: Beschreibungen und Testprogramme für die Entwicklung der Prozessormodelle sowie ausführbare Programme zur Übersetzung dieser Beschreibungen, teilweise als Evaluationsversionen.

Zu den auf der CD beigefügten Tools zählen ein VHDL-Compiler zur Synthese von Schaltungen sowie ein weiterer zur Schaltungssimulation. Beide Programme wurden als Evaluationsversion zur Verfügung gestellt. Die Synthesesoftware für Booleschen Assembler ist hingegen in der Vollversion ohne Einschränkungen enthalten. Diese Programme gestatten zumindestens die teilweise Übersetzung der textuellen Beschreibungen zu den Prozessormodellen.

Zur Übersetzung und Simulation von Assemblerprogrammen für die im Buch beschriebenen und andere (beliebige) Prozessoren wurde ein tabellengesteuerter Assembler inklusive Simulator beigefügt. Die für die Übersetzung von MPM1- und MPM2-Programmen notwendigen Tabellen sind in diesem Tool enthalten.

Zu den entsprechenden Kapiteln sind die Beschreibungen des Prozessormodells MPM1 in VHDL und Booleschem Assembler sowie MPM2 in VHDL auf der CD beigefügt. Die Codierungen in VHDL enthalten jeweils eine Verhaltensmodellierung und eine Modellierung auf Register-Transfer-Level, um sowohl die Simulation als auch die Synthese optimal zu unterstützen. Die VHDL-Beschreibungen werden durch einige Testprogramme zu den jeweiligen Prozessormodellen, in Assembler geschrieben, ergänzt.

Literaturverzeichnis

- /1.1/ *Borgmeyer, J.*: Grundlagen der Digitaltechnik. – Carl Hanser Verlag München Wien, 1997
- /1.2/ *Pernards, P.*: Digitaltechnik, 3. Aufl.. – Dr. Alfred Hüthig Verlag Heidelberg, 1992
- /1.3/ *Pernards, P.*: Digitaltechnik II, Einführung in Schaltwerke. – Dr. Alfred Hüthig Verlag Heidelberg, 1995
- /1.4/ *Schiffmann, W., Schmitz, R.*: Technische Informatik 1, Grundlagen der digitalen Elektronik. – Springer Verlag Berlin Heidelberg, 1992
- /1.5/ *Schiffmann, W., Schmitz, R.*: Technische Informatik 2, Grundlagen der Computertechnik. – Springer Verlag Berlin Heidelberg, 1992
- /1.6/ *Lehmann, G., Wunder, B., Selz, M.*: Schaltungsdesign mit VHDL. – Franzis-Verlag Poing, 1994
- /1.7/ *Armstrong, J. R., Gray, F. G.*: Structured Logic Design with VHDL. – Prentice Hall Englewood Cliffs, New Jersey, 1993.
- /1.8/ *Ten Hagen, K.*: Abstrakte Modellierung. – Springer Verlag Berlin Heidelberg, 1992
- /2.1/ *Oberschelp, W., Vossen, G.*: Rechneraufbau und Rechnerstrukturen, 5. Aufl.. – Oldenbourg Verlag München, 1992
- /2.2/ *Bähring, H.*: Mikrorechner-Systeme. – Springer Verlag Berlin Heidelberg, 1991
- /2.3/ <http://www.unicode.org>
- /2.4/ *Hennesy, J. L., Patterson, D. A.*: Computer Architecture: A Quantitative Approach, Second Edition. – Morgan Kaufmann Publishers San Francisco, 1996
- /2.5/ *Hennesy, J. L., Patterson, D. A.*: Rechnerarchitektur. – Vieweg Verlag Wiesbaden, 1994
- /3.1/ *Hennesy, J. L., Patterson, D. A.*: Computer Organization & Design: The Hardware/Software Interface, Second Edition. – Morgan Kaufmann Publishers San Francisco, 1997
- /6.1/ *Golze, U.*: VLSI-Entwurf eines RISC-Prozessors. – Vieweg Verlag Wiesbaden, 1995
- /6.2/ *Golze, U.*: Der RISC-Prozessor TOOBSIE. – Vieweg Verlag Wiesbaden, 1995
- /6.3/ *Bode, A.*: RISC-Architekturen. – Wissenschaftsverlag Mannheim, 1990
- /7.1/ *De Micheli, G.; Sami, M.*: Hardware/Software Co-Design. Kluwer Amsterdam., 1996

Sachwortverzeichnis

- Addierwerk 104
 Adreß_ALU 177
 Adreß-ALU 156
 Adreßbus 38, 71, 95
 Adressierung
 absolut 61, 75, 77, 86, 136
 absolut indirekt 62, 75, 77, 86, 136
 applikationsspezifisch 196
 Codierung (MPM1) 77
 Codierung (MPM2) 139
 direkt 60, 75, 77, 86, 136
 implizit 59, 75, 77, 86, 136
 indirekt 61
 indiziert 61, 136, 156
 MPM1-Modi 75
 MPM2-Modi 155
 MPM3-Modi 182
 registerdirekt 59
 registerindirekt 60
 relativ 62, 137, 155, 186
 unmittelbar 60, 75, 77, 86, 136
 Adreßpufferregister 87, 156
 Adreßregister 45
 Akkumulator 73, 85, 90, 131, 146
 MPM2 157
 Reset 94
 Akkumulator-Architektur 63, 72, 85
 ALU 24, 45, 48, 50, 84, 90, 100, 122
 Application Specific Instruction Set
 Processor 11, 69, 70, 193
 Application Specific Processor 189, 199
 Applikationsdomäne 190
 Arithmetik
 (n,k)-Zähler 109, 115
 Addition 104
 Addition, mehrzeilig 109
 Arithmetischer Shift 121
 BCD 101
 Carry Look Ahead Adder 107
 Division 119
 Floating-Point 100
 Halbaddierer 109
 Integer 100
 Logischer Shift 121
 Multiplikation 115
 Paralleler Addierer 106
 Ripple Carry Adder 105
 Ripple Carry Adder, mehrzeilig 110
 Sequentieller Addierer 105
 Subtraktion 111, 113
 Vergleich 124
 Volladdierer 104
 Arithmetischer Shift 121
 arithmetisch-logische Einheit
 24, 45, 48, 50, 84, 90, 94, 100, 122
 Adreß-ALU 156
 MPM2 160
 Ripple Carry ALU 122
 ASIP 11, 69, 70, 193
 ASP 189, 199
 Automat
 Mealy 22
 Medwedjew 22
 Moore 22, 93
 Befehle
 arithmetische Befehle 160
 Bedingte Sprungbefehle 78, 102, 129
 Bedingte Verzweigungsbefehle 155
 Compare 124
 Logikbefehle 102, 160
 Rotatebefehle 103, 135, 160
 Shiftbefehle 103, 135, 160
 Sprungbefehl 178
 Unterprogrammaufruf 136, 141, 152
 Verzweigungsbefehle 135, 179
 Befehlsbearbeitung 36
 Befehlsgruppen
 MPM2 133
 MPM3 182
 Befehlsklassen 55
 Befehlsphasen 170
 balanciert 175
 MPM2 162
 MPM3 186
 Befehlpufferregister 83, 84
 Befehlssatz
 Codierung 76, 137, 184
 MPM1 74, 126
 MPM2 133, 137, 139
 MPM3 181, 184
 orthogonal 75
 Betriebssystem 66
 Boolesche Variable 12
 Boolescher Assembler 25
 Bussystem 36
 CALL-Instruktion 178, 187

- Central Processing Unit 44
 CISC-Architektur..... 169
 Clocks per Instruction
 171, 173, 175, 176, 178, 179
 Co-Design 11, 70, 189
 Embedded-System..... 191
 Complex Programmable Logic Device... 19
 Control Unit..... 34, 93
 CPI.....171, 173, 175, 176, 178, 179
 CPLD 19
 CPU 44

 Data Forwarding 174, 180, 186
 Datenbus 37, 95
 MPM2 131
 Datenhazard 174, 180
 Datenregister..... 15, 45, 101
 Datenwerte
 Arithmetik 104
 ASCII-Code 54
 Bitvektoren..... 51
 Festkommaformat 53
 Festkommazahlen..... 52
 Gleitkommaformat 53
 Integer-Zahlen..... 52
 Komplementdarstellung 52, 111, 113
 UNI-Code..... 54
 Vorzeichen 52, 113
 Zweierkomplement 52, 111, 113, 122
 DEA 21, 26
 Deadlock..... 177
 Decode-Phase 57, 76, 77, 79
 MPM1 84
 MPM2 140, 161
 MPM3 186
 deterministischer endlicher Automat 21, 26
 DFA 21, 26
 disjunktive Normalform..... 14
 DNF 14

 Echtzeitbedingungen..... 191, 198
 Eingabe-/Ausgabebereich 43
 Embedded-System 190
 Entwurfsebenen 28
 Entwurfssichten 28
 Execute-Phase..... 57, 79, 178
 MPM1 88
 MPM2 141
 MPM3 186

 Fetch-Phase..... 55, 79, 175, 177, 178
 MPM1 82
 MPM2 140, 148
 MPM3 186
 Field Programmable Gate Array.....20
 Finite State Machine Syntax.....26
 Flags46, 94
 Borrow Carry111
 Carry
 73, 74, 85, 94, 101, 102, 103, 112,
 123, 131, 135
 Halfcarry101
 MPM1126
 MPM2131, 159, 161
 Negative101, 126, 131
 Overflow101, 102, 114, 123, 131
 Parity195
 Zero.....73, 74, 94, 102, 126, 131
 Flipflop16
 Charakteristische Gleichung16
 FPGA.....20
 FSM.....26

 General-Purpose-Register.....177, 181
 Gleitkommaformat53

 Hardwarebeschreibungssprachen25
 Hardware-Software-Co-Design .11, 70, 189
 Harvard-Modell72, 177
 Hazard173
 Datenhazard174, 180
 Kontrollflußhazard.....175, 178
 RAW174
 Strukturhazard.....173, 175, 177, 181
 HDL.....25

 I/O-Prozessoren44
 Indexregister.....131, 146, 156
 MPM2157
 Input/Output-Bereich.....43
 Instruction Pointer .. *Siehe* Programmzähler
 Instruction Set Architektur192
 Interrupt47, 64, 141, 196
 Enable Flag65, 67, 131, 135, 142
 MPM2141, 146, 148
 pending65
 Prioritäten65
 Reentrant.....149
 Request Controller68
 RTI-Befehl143, 151
 Service-Routine65, 150, 198
 Softwareinterrupt67
 Stackoperationen.....151
 Trap.....67

- Vektor MPM2 130
 Vektornummer 65, 68
 ISA 192
- Jitter 198
- Komplexe Instruktionen 169
 Kontrollflußhazard 175, 178
 Branch Prediction 179
 Delay-Branch 179
 Look-Ahead-Resolution 179
- Linkregister 187
 Load/Store-Architektur 177, 181
 Load-Phase 57, 76, 79, 175, 177
 MPM1 85
 MPM2 140
 MPM3 186
 Load-Store-Architektur 64
 Logikfunktionen
 AND 102
 Einerkomplement 102
 OR 102
 XOR 102
 Logischer Shift 121
 Lokalitätsprinzip 62
 Look up Table 14
 LUT 14
- Memory 43
 MPSC 176
 Multiplexer 15, 122
 Multiport-Access 177
- Operationswerk 24
- parallele Schnittstelle 44
 PC
 Reset 94
- Phasen
 Decode 57, 76, 77, 79, 140
 Execute 57, 79, 141
 Fetch 55, 65, 79, 82, 140
 Load 57, 76, 79, 140
 Write Back 57, 79, 141
- Phasen der Befehlsbearbeitung 55
 MPM2 140
- Phasenpipelining 36, 170, 174
 Ausnahmebehandlung 187
 MPM2 172
 MPM3 180
 Wartezyklen 173
- Pipeline Stalls 175, 176, 180
 Pipelining 170
 MPSC 176
 Wartezyklen 175, 180
 Pipeliningstufe 171
 PLD 12
 Profiling 193
 Program Counter *Siehe* Programmzähler
 Programmierbare Logikbausteine 12
 Programmiermodell 45, 69
 MPM2 131
 Programmzähler
 39, 45, 46, 73, 83, 85, 89, 129, 155
 Reset 94
 Prozessorklassifizierung 63, 177
- RAM 43
 Random Access Read Write Memory 43
 RAW-Hazard 174, 180
 Read Only Memory 43
 Read-After-Write-Hazard 180
 Rechenwerk 45, 48, 50, 89, 100
 Registerfile 49
 stackorientiert 49
 Register-Architektur 63
 Registermodell 73
 MPM2 131
 Register-Speicher-Architektur 64
 Register-Transfer-Ebene 29, 69, 81, 98
 Reset
 MPM1 93, 94
 MPM2 147, 154
 Resetvektor 94, 130, 147
 Responsive System 191
 RISC-Architektur 170, 172, 180
 ROM 43
 RTI-Instruktion 187
 RTL 29, 69, 81, 98
 RTS-Instruktion 187
- Schaltnetz 12
 Schaltvariable 12
 Schaltwerk 13
 einfach 22
 komplex, kooperierend 22
 zustandsorientiert 22
- Sequentialität 100
 serielle Schnittstelle 44
 Simple Programmable Logic Device 17
 Speicher 43
 Speichermodell 69, 71, 129
 SPLD 17

Stack	47, 129	Universalrechner.....	33
Adressierung	48	Unterbrechungsanforderung	47, 64, 196
Hardwarestack.....	47	Unterprogramm sprung	187
MPM2	153		
Overflow	48, 132, 152	Verzweigungsbe fehle	195
Softwarestack	47	VHDL.....	28
Speicherbereich MPM2.....	130	Architecture	30
Stackpointer	45, 47, 146, 153	Configuration	30
Stack-Architektur.....	63	Entity.....	30
Stackpointer	131	Package	30
Statusregister.....	45, 46, 73, 85, 90, 94, 101	Process	31
MPM2	131, 159	strukturelle Sicht	28, 167
Reset.....	94, 159	Verhaltenssicht.....	28, 80, 98, 143
Steuerbus.....	38	Volladdierer	104
Steuerregister	45	Von-Neumann-Modell	177
Steuerwerk	24, 34, 89, 90, 93, 100	Von-Neumann-Rechner.....	33, 146
MPM2	162	Zentraleinheit.....	33
Strukturhazard.....	173, 175, 177, 181		
Systembus		Write-Back-Phase.....	57, 79, 178
Handshake	40	MPM1	90
semisynchron	41	MPM2	141
synchron	40	MPM3	186
Systembusschnittstelle	38	Y-Diagramm.....	28
Trap.....	67	Zentraleinheit.....	44
		Zero-Delay.....	191