

# Inhaltsverzeichnis

<b>1</b>	<b>Einführung</b>	<b>1</b>
1.1	Geschichtliche Entwicklung der Mikroprozessortechnik . . . . .	1
1.2	Stand und Entwicklungstempo der Mikroprozessortechnik . . . . .	3
1.3	Grundbestandteile eines Mikrorechnersystems . . . . .	4
1.4	Aufgaben und Testfragen . . . . .	6
<b>2</b>	<b>Informationseinheiten und Informationsdarstellung</b>	<b>7</b>
2.1	Bits, Tetraden, Bytes und Worte . . . . .	7
2.2	Die Interpretation von Bitmustern . . . . .	8
2.3	Zahlensysteme . . . . .	9
2.4	Die binäre Darstellung von Zahlen . . . . .	10
2.4.1	Vorzeichenlose ganze Zahlen . . . . .	11
2.4.2	Vorzeichenbehaftete ganze Zahlen (Zweierkomplement-Darstellung) . . . . .	11
2.4.3	Festkommazahlen . . . . .	13
2.4.4	Gleitkommazahlen . . . . .	14
2.5	Aufgaben und Testfragen . . . . .	18
<b>3</b>	<b>Halbleiterbauelemente</b>	<b>19</b>
3.1	Diskrete Halbleiterbauelemente . . . . .	19
3.1.1	Dotierte Halbleiter . . . . .	19
3.1.2	Feldeffekttransistoren . . . . .	19
3.2	Integrierte Schaltkreise (Integrated Circuits) . . . . .	21
3.2.1	Allgemeines . . . . .	21
3.2.2	Schaltkreisfamilien . . . . .	23
3.2.3	TTL-Bausteine . . . . .	24
3.2.4	CMOS-Bausteine . . . . .	25
3.2.5	Weitere Schaltkreisfamilien . . . . .	27
3.2.6	Logische Verknüpfungen und Logische Schaltglieder . . . . .	27
3.3	Aufgaben und Testfragen . . . . .	30
<b>4</b>	<b>Speicherbausteine</b>	<b>31</b>
4.1	Allgemeine Eigenschaften . . . . .	31
4.2	Read Only Memory (ROM) . . . . .	34
4.2.1	Masken-ROM (MROM) . . . . .	34
4.2.2	Programmable ROM (PROM) . . . . .	35
4.2.3	Erasable PROM (EPROM) . . . . .	36

4.2.4	EEPROM und Flash-Speicher . . . . .	37
4.3	Random Access Memory (RAM) . . . . .	38
4.3.1	Statisches RAM (SRAM) . . . . .	38
4.3.2	Dynamisches RAM (DRAM) . . . . .	40
4.4	Magnetoresistives RAM und Ferroelektrisches RAM . . . . .	52
4.5	Aufgaben und Testfragen . . . . .	54
<b>5</b>	<b>Ein- und Ausgabe</b> . . . . .	<b>56</b>
5.1	Allgemeines . . . . .	56
5.2	Eingabeschaltung, Ausgabeschaltung . . . . .	56
5.3	Ein-/Ausgabe-Steuerung von Bausteinen und Geräten . . . . .	58
5.3.1	Aufbau von Bausteinen und Geräten mit Ein-/Ausgabe-Steuerung . . . . .	58
5.3.2	Fallbeispiel: Der programmierbare Ein-/Ausgabebaustein 8255 . . . . .	59
5.4	Aufgaben und Testfragen . . . . .	61
<b>6</b>	<b>Systembus und Adressverwaltung</b> . . . . .	<b>62</b>
6.1	Busaufbau . . . . .	62
6.1.1	Warum ein Bus? . . . . .	62
6.1.2	Open-Drain-Ausgänge . . . . .	63
6.1.3	Tristate-Ausgänge . . . . .	64
6.1.4	Bustreiber . . . . .	65
6.1.5	Synchrone und asynchrone Busse . . . . .	67
6.1.6	Busdesign . . . . .	68
6.1.7	Busvergabe bei mehreren Busmastern . . . . .	69
6.2	Busanschluss und Adressverwaltung . . . . .	70
6.2.1	Allgemeines zu Adressen und ihrer Dekodierung . . . . .	70
6.2.2	Adressdekodierung von Ein-/Ausgabebausteinen . . . . .	71
6.2.3	Adressdekodierung von Speicherbausteinen . . . . .	76
6.2.4	Big-Endian- und Little-Endian-Byteordnung . . . . .	80
6.3	Chipsätze moderner PCs . . . . .	81
6.4	Aufgaben und Testfragen . . . . .	85
<b>7</b>	<b>Einfache Mikroprozessoren</b> . . . . .	<b>87</b>
7.1	Die Ausführung des Maschinencodes . . . . .	87
7.2	Interner Aufbau eines Mikroprozessors . . . . .	89
7.2.1	Registersatz . . . . .	89
7.2.2	Steuerwerk . . . . .	91
7.2.3	Operationswerk (Rechenwerk) . . . . .	93
7.2.4	Adresswerk und Adressierungsarten . . . . .	95
7.2.5	Systembus-Schnittstelle . . . . .	99
7.3	CISC-Architektur und Mikroprogrammierung . . . . .	100
7.4	RISC-Architektur . . . . .	101
7.5	Programmierung von Mikroprozessoren . . . . .	103
7.5.1	Maschinenbefehlssatz . . . . .	103
7.5.2	Maschinencode und Maschinenprogramme . . . . .	105
7.5.3	Assemblersprache und Compiler . . . . .	106
7.5.4	Hardware-Software-Schnittstelle (Instruction Set Architecture) . . . . .	107

7.6	Reset und Boot-Vorgang . . . . .	108
7.7	Erganzung: Hilfsschaltungen . . . . .	109
7.7.1	Taktgenerator . . . . .	109
7.7.2	Einschaltverzogerung . . . . .	109
7.8	Aufgaben und Testfragen . . . . .	109
<b>8</b>	<b>Besondere Betriebsarten</b> . . . . .	<b>111</b>
8.1	Interrupts (Unterbrechungen) . . . . .	111
8.1.1	Das Problem der asynchronen Service-Anforderungen . . . . .	111
8.1.2	Das Interruptkonzept . . . . .	112
8.1.3	Interrupt-Behandlungsroutinen . . . . .	112
8.1.4	Aufschaltung und Priorisierung von Interrupts . . . . .	113
8.1.5	Vektorisierung und Maskierung von Interrupts, Interrupt-Controller . . . . .	114
8.2	Ausnahmen (Exceptions) . . . . .	116
8.3	Direct Memory Access (DMA) . . . . .	116
8.4	Aufgaben und Testfragen . . . . .	118
<b>9</b>	<b>Beispielarchitekturen</b> . . . . .	<b>119</b>
9.1	Die CPU08 von Freescale . . . . .	119
9.1.1	Übersicht . . . . .	120
9.1.2	Der Registersatz . . . . .	121
9.1.3	Der Adressraum . . . . .	123
9.1.4	Die Adressierungsarten . . . . .	123
9.1.5	Der Befehlssatz . . . . .	127
9.1.6	Unterprogramme . . . . .	129
9.1.7	Reset und Interrupts . . . . .	130
9.1.8	Codebeispiele . . . . .	133
9.2	Die MSP430CPU von Texas Instruments . . . . .	141
9.2.1	Übersicht . . . . .	141
9.2.2	Der Registersatz . . . . .	142
9.2.3	Der Adressraum . . . . .	143
9.2.4	Die Adressierungsarten . . . . .	144
9.2.5	Der Befehlssatz . . . . .	145
9.2.6	Reset und Interrupts . . . . .	147
9.2.7	Unterstützung für die ALU: Der Hardware-Multiplizierer . . . . .	150
9.2.8	Codebeispiele . . . . .	151
9.3	Der ARM Cortex-M3 . . . . .	156
9.3.1	Historie der ARM- und Cortex-Prozessoren . . . . .	156
9.3.2	Übersicht . . . . .	158
9.3.3	Der Registersatz des Cortex-M3 . . . . .	158
9.3.4	Der Adressraum und Adressierungsarten . . . . .	160
9.3.5	Der Befehlssatz . . . . .	161
9.3.6	Reset, Exceptions und Interrupts . . . . .	165
9.3.7	Schutzmechanismen des Cortex-M3 . . . . .	166
9.3.8	Erstellung von Software . . . . .	167
9.4	Kurzer Vergleich der drei Beispielarchitekturen . . . . .	170

9.5	Aufgaben und Testfragen	171
<b>10</b>	<b>Speicherverwaltung</b>	<b>173</b>
10.1	Virtueller Speicher und Paging	173
10.2	Speichersegmentierung	177
10.3	Caching	180
10.3.1	Warum Caches?	180
10.3.2	Strukturen und Organisationsformen von Caches	183
10.3.3	Ersetzungsstrategien	187
10.3.4	Aktualisierungsstrategien	187
10.4	Fallstudie: Intel Pentium 4 (IA-32-Architektur)	189
10.4.1	Privilegierungsstufen	189
10.4.2	Speichersegmentierung, Selektoren und Deskriptoren	191
10.4.3	Paging	195
10.4.4	Kontrolle von E/A-Zugriffen	197
10.4.5	Caches	197
10.4.6	Der Aufbau des Maschinencodes	198
10.5	Aufgaben und Testfragen	201
<b>11</b>	<b>Skalare und superskalare Architekturen</b>	<b>203</b>
11.1	Skalare Architekturen und Befehls-Pipelining	203
11.2	Superskalare Architekturen	209
11.2.1	Mehrfache parallele Hardwareeinheiten	209
11.2.2	Ausführung in geänderter Reihenfolge	212
11.2.3	Register-Umbenennung	214
11.2.4	Pipeline-Länge, spekulative Ausführung	214
11.2.5	VLIW-Prozessoren	216
11.2.6	Hyper-Threading	216
11.2.7	Prozessoren mit mehreren Kernen	218
11.3	Fallbeispiel: Core Architektur der Intel-Prozessoren	222
11.3.1	Der 64-Bit-Registersatz	222
11.3.2	Die Entwicklung bis zu Pentium 4	223
11.3.3	Die Mikroarchitektur der Core Prozessoren	227
11.4	Fallbeispiel: IA-64 und Itanium-Prozessor	231
11.5	Aufgaben und Testfragen	235
<b>12</b>	<b>Energieeffizienz von Mikroprozessoren</b>	<b>237</b>
12.1	Was ist Energieeffizienz und warum wird sie gebraucht?	237
12.2	Leistungsaufnahme von integrierten Schaltkreisen	238
12.2.1	Verminderung der Leistungsaufnahme	241
12.3	Das Advanced Control and Power Interface (ACPI)	242
12.4	Praktische Realisierung von energieeffizienten Architekturen und Betriebsarten	243
12.4.1	Mikrocontroller	243
12.4.2	PC-Prozessoren	244
12.4.3	Prozessoren für Subnotebooks	247
12.5	Aufgaben und Testfragen	248

<b>13 Single Instruction Multiple Data (SIMD)</b>	<b>249</b>
13.1 Grundlagen	249
13.2 Fallbeispiel: SIMD bei Intels IA-32-Architektur	250
13.2.1 Die MMX-Einheit	251
13.2.2 Die SSE-, SSE2-, SSE3- und SSE4-Befehle	254
13.3 Aufgaben und Testfragen	256
<b>14 Mikrocontroller</b>	<b>258</b>
14.1 Allgemeines	258
14.2 Typische Baugruppen von Mikrocontrollern	259
14.2.1 Mikrocontrollerkern (Core)	259
14.2.2 Busschnittstelle	260
14.2.3 Programmspeicher	260
14.2.4 Datenspeicher	261
14.2.5 Ein-/Ausgabeschnittstellen (Input/Output-Ports)	261
14.2.6 Zähler/Zeitgeber (Counter/Timer)	262
14.2.7 Analoge Signale	267
14.2.8 Interrupt-System	269
14.2.9 Komponenten zur Datenübertragung	270
14.2.10 Bausteine für die Betriebssicherheit	273
14.2.11 Energieeffizienz	274
14.2.12 Die JTAG-Schnittstelle	274
14.3 Software-Entwicklung	277
14.3.1 Einführung	277
14.3.2 Programmstruktur	278
14.3.3 Header-Dateien	282
14.3.4 Die Übertragung des Programmes auf das Zielsystem	284
14.3.5 Programmtest	286
14.3.6 Integrierte Entwicklungsumgebungen	288
14.4 Fallbeispiel: Der MSP430 von Texas Instruments	290
14.4.1 Der Watchdog Timer+	290
14.4.2 Digitale Ein- und Ausgänge	291
14.4.3 Der Zähler/Zeitgeber Timer_A	293
14.4.4 Der 10-Bit-Analog/Digital-Wandler ADC10	297
14.4.5 Serielle Schnittstellen	299
14.5 Aufgaben und Testfragen	301
<b>15 Digitale Signalprozessoren</b>	<b>302</b>
15.1 Digitale Signalverarbeitung	302
15.2 Architekturmerkmale	305
15.2.1 Kern	305
15.2.2 Peripherie	308
15.3 Fallbeispiel: Die DSP56800-Familie von Freescale	308
15.3.1 Kern der DSP56800	309
15.3.2 DSP-Peripherie am Beispiel des DSP56F801	313
15.4 Aufgaben und Testfragen	314

<b>Lösungen zu den Aufgaben und Testfragen</b>	<b>315</b>
<b>Literaturverzeichnis</b>	<b>326</b>
<b>Sachwortverzeichnis</b>	<b>329</b>