

Inhaltsverzeichnis

1	Einführung, Bauformen und Technologien	1
1.1	Begriffsbestimmung und Beispiele.....	1
1.2	Systemkategorien	3
1.3	Typischer Aufbau	4
1.4	Bauformen von Eingebetteten Systemen	8
1.5	Technologien von Eingebetteten Systemen	12
1.5.1	Kunden- bzw. anwendungsspezifische Hardware	12
1.5.2	Feldprogrammierbare Bausteine	15
1.6	Verteilte Systeme.....	20
1.7	Zusammenfassung	22
2	Mikroprozessor-Grundlagen	23
2.1	Evolution der Mikroprozessoren.....	23
2.2	Mikroprozessoren in Eingebetteten Systemen.....	24
2.3	Mikroprozessor-Grundlagen	25
2.3.1	Mikroprozessor-Architekturen.....	25
2.3.2	Ein- und Ausgabe durch Befehle und Interrupts.....	28
2.3.3	Speicher-Systeme	31
2.3.4	Wozu brauchen wir Caches?	31
2.3.5	Hauptspeicher	36
2.3.6	Festwertspeicher (ROM)	38
2.3.7	Befehls-Verarbeitungsmethoden und Pipelining	39
2.3.8	Performanz und Energieverbrauch	43
2.3.9	Leistungsverbrauchs-Steuerung (Power Management).....	46
2.3.10	Ein-Ausgabe-Geräte und Schnittstellen	48
2.4	Mikrokontroller.....	49
2.4.1	Niedrigpreis-Mikrokontroller	49
2.4.2	Mikrokontroller höherer Leistung	50
2.5	Multi-Core- und Mehrprozessorsysteme	51
2.6	Mikroprozessor-Familien	52
2.7	Betriebssysteme	62

2.7.1	Wann kann auf ein Betriebssystem verzichtet werden?	62
2.7.2	Konzepte von Betriebssystemen	65
2.7.3	Prozesse	65
2.7.4	Aufgaben und Schichtenmodell eines Betriebssystems	68
2.7.5	Arten von Betriebssystemen	69
2.7.6	Strukturen von Betriebssystemen	70
2.7.7	Echtzeitbetriebssysteme und Echtzeitsysteme	72
2.7.8	Zeitablaufplanung in Echtzeitbetriebssystemen	74
2.7.9	Prioritätsumkehr und Prioritätsvererbung	79
2.7.10	Betriebssystem-Beispiele für Eingebettete Systeme	81
2.8	Zusammenfassung	84
3	Kommunikation, Busse und Netzwerke	85
3.1	Das ISO-OSI-Referenzmodell	85
3.2	Der parallele Bus	89
3.2.1	Schema eines Bustreibers	90
3.2.2	Bus-Kommunikation	91
3.2.3	Bus-Zeitablaufpläne	92
3.2.4	Optimierte Übertragungsmodi: Burst und Pipelining	94
3.2.5	Direkter Speicherzugriff (DMA) über einen Bus	95
3.2.6	Multiprozessor- und Multibus-Systeme	96
3.2.7	Der Arbiter	97
3.2.8	Bekannte Bus-Protokolle	97
3.3	Netzwerke	97
3.3.1	Kommunikationsmodi	98
3.3.2	Netzwerk-Topologien	98
3.3.3	Datenformatierung	101
3.4	Busähnliche Netzwerke oder serielle Busse	102
3.4.1	Der I^2C -Bus	103
3.4.2	Der CAN-Bus	105
3.4.3	FlexRay, LIN und MOST	107
3.4.4	Profibus (Process Field Bus)	108
3.4.5	Weitere Verbindungsnetzwerke für verteilte Systeme	110
3.5	Synchronisierung	111
3.5.1	Kommunikationsprimitive	112
3.5.2	Synchronisierung von Prozessorelementen	114
3.6	Zusammenfassung	116
4	Entwicklungsmethodik	117
4.1	Die Produktivitätslücke	117
4.2	Anforderungen	119
4.2.1	Nichtfunktionale Anforderungen	119

4.2.2	Lastenheft	123
4.3	Der Beginn einer Entwicklung	125
4.3.1	Der Architekturbegriff	125
4.3.2	Pflichtenheft	126
4.3.3	Spezifikation	127
4.4	Entwurfsmethoden von Software-Systemen	129
4.4.1	Das Wasserfallmodell	129
4.4.2	Das Spiralmodell	130
4.4.3	Agile Software-Entwicklungsmethoden	131
4.4.4	Die Projekt-Entwicklungsmethode Scrum	132
4.5	Software-Entwicklung für Eingebettete Systeme	134
4.5.1	Programmentwicklung	135
4.5.2	Entwickeln von Klassendiagrammen mit CRC-Karten	135
4.5.3	Entwurfsmuster	137
4.5.4	Programm-Modelle	138
4.5.5	Compiler	140
4.5.6	Programm-Optimierungen	141
4.6	Hardware-Entwicklungsmethodik	146
4.6.1	Die Abstraktionsebenen	146
4.6.2	Evolution der Entwicklungsmethoden	150
4.7	Modellieren von Systemen	158
4.7.1	Was sind Modelle?	158
4.7.2	Modelle auf System- und algorithmischer Ebene: Berechnungsmodelle ..	160
4.7.3	Unified Modeling Language (UML)	172
4.7.4	Transaction Level-Modellierung (TLM)	174
4.7.5	Modellieren auf RT-Ebene	179
4.7.6	Modelle auf Logik-Ebene	183
4.8	Plattformbasierter Entwurf	183
4.9	Die Modellbasierte Entwicklungsmethode	188
4.9.1	Synthese	189
4.9.2	Synthese auf Systemebene	190
4.10	Zeitverhalten und Performanzabschätzungen	195
4.10.1	TLM-basierte Performanz-Abschätzung	195
4.10.2	Performanz-Abschätzung aus Compiler-optimiertem Maschinencode	198
4.11	Verifikation, Simulation und Validierung	208
4.11.1	Simulation	209
4.11.2	Formale Verifikation	216
4.11.3	Werkzeuge für Modellierung und Simulation	218
4.12	Test	221
4.12.1	Begriffsbestimmungen, Black-Box- und White-Box-Test	221
4.12.2	Ein klassisches Fehlermodell	222
4.12.3	Testmuster	223

4.12.4	JTAG Boundary-Scan	223
4.13	Zusammenfassung	226
5	Beschreibungssprachen für den Systementwurf	227
5.1	VHDL – Eine Hardware-Beschreibungssprache	228
5.1.1	Grundlegender Aufbau	230
5.1.2	Das Sprachkonzept	233
5.1.3	Die Schaltungsbeschreibung	234
5.1.4	Signale und Datentypen	238
5.1.5	Zuweisungen und die neunwertige Standard Logik	242
5.1.6	Operationen	245
5.1.7	Die eventgesteuerte VHDL-Simulation und der Delta-Zyklus	248
5.1.8	Der VHDL-Prozess	250
5.1.9	Beispiele einfacher Prozessbeschreibungen	255
5.1.10	Komponenten mit größeren Datenbreiten und generische Komponenten	258
5.1.11	Konfigurationsanweisungen	262
5.1.12	Der VHDL-Prozess als Beschreibung für Schaltwerke	263
5.1.13	Beispiel eines Simulationstreibers in VHDL	269
5.1.14	VHDL-Attribute	271
5.1.15	Unterprogramme und Packages	272
5.1.16	Typ-Konvertierungen	275
5.1.17	Die Assert-Anweisung	278
5.1.18	Simulationsbeispiel für ein einfaches Zweiprozessorsystem	279
5.1.19	Zusammenfassung	284
5.2	Die System-Beschreibungssprache SystemC	284
5.2.1	Grundlagen von SystemC	286
5.2.2	Beispiel eines SystemC-Moduls	289
5.2.3	Simulationssemantik	298
5.2.4	Transaction-Level-Modellierung mit SystemC	300
5.2.5	RTL-Modellierung mit SystemC	301
5.2.6	Zusammenfassung	302
6	Sensornetzwerke	303
6.1	Drahtlose Sensornetzwerke	303
6.2	Kommunikation im Sensornetzwerk	306
6.2.1	Der Protokoll-Stapel (Protocol Stack) von drahtlosen Sensornetzwerken	308
6.2.2	Medium Access Control (MAC)	311
6.2.3	Physikalische Schicht von drahtlosen Sensornetzwerken	317
6.3	Der Sensorknoten	317
6.4	Kommunikationsstandards für drahtlose Netzwerke	321
6.5	Zusammenfassung	322

7	Software-Synthese und High-Level-Synthese	323
7.1	Herausforderungen der Software-Entwicklung	323
7.2	Software-Synthese von Eingebetteten Systemen	324
7.3	High-Level-Synthese	332
7.3.1	Einführung in die High-Level-Synthese	332
7.3.2	Eingabe und Ausgabe eines High-Level-Synthese-Werkzeugs	333
7.3.3	Bewertung von elektronischen Schaltkreisen	336
7.4	Die wesentlichen Schritte der High-Level-Synthese.....	339
7.4.1	Allokierung.....	339
7.4.2	Die Komponentenbibliothek	340
7.5	Zeitablaufplanung (Scheduling)	341
7.5.1	Einführung in die Zeitablaufplanung: Der Sequenzgraph	341
7.5.2	Zeitablaufplanung ohne Ressource-Schranken.....	344
7.5.3	Zeitablaufplanung unter Ressourcenschranken	348
7.5.4	Einige besondere Verfahren der Zeitablaufplanung.....	360
7.5.5	Zusammenfassung des Abschnitts Zeitablaufplanung	364
7.6	Ressourcen-Bindung	364
7.6.1	Kompatibilitäts- und Konfliktgraph.....	365
7.6.2	Bindung von Speicher-Ressourcen (Registern)	368
7.6.3	Die Graphen-Partitionierungs-Methode	371
7.6.4	Zuweisung von Verbindungselementen und Erstellung der Netzliste	372
7.7	Steuerwerksynthese	379
7.7.1	Das mikrocodebasierte Steuerwerk	379
7.7.2	Das hartverdrahtete Steuerwerk	380
7.8	Zusammenfassung	382
	Literaturverzeichnis	383
	Index	391