

Inhaltsverzeichnis

1	Einführung, Systeme, Bauformen und Technologien	1
1.1	Begriffsbestimmung und Beispiele.....	1
1.2	Systemkategorien.....	3
1.3	Typischer Aufbau	4
1.3.1	Beispiele für Sensoren und Aktoren	7
1.4	Cyber-Physische Systeme (Cyber Physical Systems).....	8
1.5	Systeme mit Künstlicher Intelligenz (KI-Systeme)	11
1.5.1	Anwendungen von KI-Systemen	13
1.6	Verteilte Systeme.....	17
1.7	Bauformen von Eingebetteten Systemen	18
1.7.1	Prozessorarten	21
1.8	Schaltkreise für Eingebettete Systeme.....	23
1.8.1	Herstellerkonfigurierte Schaltkreise.....	23
1.8.2	Anwenderkonfigurierbare Schaltkreise	24
1.9	Zusammenfassung	32
2	Entwicklungsmethodik	33
2.1	Kundenanforderungen und Spezifikation	34
2.1.1	Nichtfunktionale Anforderungen	34
2.1.2	Lastenheft.....	37
2.1.3	Pflichtenheft	39
2.1.4	Spezifikation	40
2.2	Der Beginn einer Entwicklung	41
2.2.1	Der Architekturbegriff.....	41
2.3	Hardware/Software-Co-Entwurf.....	43
2.4	Software-Entwicklung.....	44
2.4.1	Das Wasserfallmodell	44
2.4.2	Das Spiralmodell	46
2.4.3	Agile Software-Entwicklungsmethoden	46
2.4.4	Die Projekt-Entwicklungsmethode Scrum	47
2.4.5	Programmentwicklung	49
2.4.6	Entwickeln von Klassendiagrammen mit CRC-Karten	50

2.4.7	Entwurfsmuster	52
2.5	Hardware-Entwicklungsmethodik	53
2.5.1	Die Produktivitätslücke	53
2.5.2	Die Abstraktionsebenen	55
2.5.3	Evolution der Entwicklungsmethoden	58
2.6	Plattformbasierter Entwurf	66
2.7	Transaction-Level-Modellierung (TLM)	68
2.8	Die Modellbasierte Entwicklungsmethode	69
2.9	Plattformen und Bibliotheken für den Entwurf von DNNs	71
2.10	Berücksichtigung des Energiebedarfs bei der Entwicklung	72
2.11	Zusammenfassung	75
3	Modelle	77
3.1	Definition eines Modells	77
3.2	Programm-Modelle	78
3.3	Modellkategorien	79
3.4	Modelle auf System- und algorithmischer Ebene: Berechnungsmodelle ...	81
3.4.1	Prozessbasierte Berechnungsmodelle und -Netzwerke	82
3.4.2	Kahn-Prozess-Netzwerke (KPN)	83
3.4.3	Datenfluss-Netzwerke	85
3.4.4	Prozess-Kalkuli (Process Calculi)	87
3.5	Zustandsbasierte Modelle	88
3.5.1	Petri-Netze	88
3.5.2	StateCharts	90
3.5.3	Prozess-Zustandsmaschinen	93
3.6	Unified Modeling Language (UML)	94
3.7	Transaction-Level-Modellierung (TLM)	96
3.7.1	Modellieren der Kommunikation mit TLMs	98
3.7.2	Das Netzwerk-TLM	99
3.7.3	Protokoll-TLM und Bus-zyklusgenaues Modell	100
3.8	Modellieren auf RT-Ebene	100
3.8.1	Das grundlegende Verhaltensmodell auf RT-Ebene	101
3.8.2	Das Strukturmodell auf RT-Ebene (PCAM)	103
3.9	Modelle auf Logik-Ebene	104
3.10	Zusammenfassung	105
4	Eingebettete KI-Systeme	107
4.1	Grundlagen künstlicher Neuronaler Netze	108
4.1.1	Feed-Forward Netze	113

4.1.2	Tiefe Neuronale Netzwerke (Deep Neural Networks)	119
4.1.3	Zeitlich veränderliche Datenreihen - Sequentielle Daten	120
4.1.4	Rekurrente Neuronale Netze	121
4.1.5	Long Short-Term Memory LSTM	125
4.1.6	Temporale Faltungs-Netze – TCN	128
4.1.7	Residual Netzwerke: ResNets	130
4.2	Maschinelles Lernen oder Training von NN	132
4.2.1	Feed-Forward Lernen	136
4.2.2	Backpropagation	137
4.2.3	Maschinelles Lernen oder die Vorgehensweise beim Anlernen von Neuronalen Netzwerken	138
4.3	Training und Inferenz	139
4.3.1	Prominente DNNs auf der ILSVRC	141
4.4	Hardware für Eingebettete KI-Systeme	144
4.4.1	Hardware-Kategorien für Eingebettete KI-Systeme	144
4.4.2	Optimierung von Neuronalen Netzen	144
4.4.3	Computer-Rechenleistung zum Anlernen von Neuronalen Netzen	144
4.4.4	Reduzierung der Komplexität von DNNs	146
4.4.5	Hardware-Architekturen für DNN-Rechenbeschleuniger	151
4.4.6	Eine klassische Anwendung von KI: Autonomes Fahren	154
4.4.7	Die Elektrik/Elektronik-Architektur	155
4.4.8	Das elektronische Steuersystem	156
4.4.9	Autonomes Fahren: Beispiel für die Implementierung	158
4.4.10	Elektronik-Hardware für KI-Systeme	159
4.4.11	Ein „Always on KWS-System“: UltraTrail	162
4.5	Zusammenfassung	165
5	Beschreibungssprachen für den Systementwurf	167
5.1	VHDL – Eine Hardware-Beschreibungssprache	168
5.1.1	Grundlegender Aufbau	170
5.1.2	Das Sprachkonzept	173
5.1.3	Die Schaltungsbeschreibung	174
5.1.4	Signale und Datentypen	178
5.1.5	Zuweisungen und die neunwertige Standard Logik	182
5.1.6	Operationen	185
5.1.7	Die eventgesteuerte VHDL-Simulation und der Delta-Zyklus	187
5.1.8	Der VHDL-Prozess	190
5.1.9	Beispiele einfacher Prozessbeschreibungen	194
5.1.10	Generische Komponenten mit größeren Datenbreiten	198
5.1.11	Konfigurationsanweisungen	202
5.1.12	Der VHDL-Prozess als Beschreibung für Schaltwerke	203
5.1.13	Beispiel eines Simulationstreibers in VHDL	208
5.1.14	VHDL-Attribute	210
5.1.15	Unterprogramme und Packages	212
5.1.16	Typ-Konvertierungen	215

5.1.17	Die Assert-Anweisung	218
5.1.18	Simulationsbeispiel für ein Zweiprozessorsystem	218
5.1.19	Entwurf energiesparsamer Hardwaresysteme mit VHDL	223
5.1.20	Implementierung von Power Gating mit UPF und VHDL	224
5.1.21	Zusammenfassung	225
5.2	Die System-Beschreibungssprache SystemC	225
5.2.1	Grundlagen von SystemC	227
5.2.2	Beispiel eines SystemC-Moduls	230
5.2.3	Simulationssemantik	239
5.2.4	Transaction-Level-Modellierung mit SystemC	240
5.2.5	RTL-Modellierung mit SystemC	241
5.2.6	Zusammenfassung	242
6	Eingebettete Software	243
6.1	Betriebssysteme	243
6.1.1	Wann kann auf ein Betriebssystem verzichtet werden?	243
6.1.2	Konzepte von Betriebssystemen	246
6.1.3	Prozesse	246
6.1.4	Aufgaben und Schichtenmodell eines Betriebssystems	249
6.1.5	Arten von Betriebssystemen	251
6.1.6	Strukturen von Betriebssystemen	252
6.1.7	Echtzeitbetriebssysteme und Echtzeitsysteme	254
6.1.8	Zeitablaufplanung in Echtzeitbetriebssystemen	255
6.1.9	Prioritätsumkehr	260
6.1.10	Prioritätsvererbung	261
6.1.11	Betriebssystem-Beispiele für Eingebettete Systeme	262
6.2	Compiler	265
6.3	Programm-Optimierungen	266
6.3.1	Programm-Optimierung auf höherer Ebene	266
6.3.2	Minimierung des Energiebedarfs in Programmen	268
6.3.3	Optimierung der Programmgröße	269
6.4	Performanzabschätzungen und Zeitverhalten (Timing Analyse)	270
6.4.1	TLM-basierte Performanz-Abschätzung	270
6.4.2	Performanz-Abschätzung aus Compiler-optimiertem Maschinencode	273
6.5	Software-Synthese	282
6.5.1	Herausforderungen der Software-Entwicklung	282
6.5.2	Software-Synthese von Eingebetteten Systemen	283
6.5.3	Code-Generierung in der Software-Synthese	285
6.6	Zusammenfassung	290

7	Hardware-Synthese	291
7.1	Synthese auf verschiedenen Abstraktionsebenen	291
7.2	Synthese auf Systemebene nach Gajski	292
7.3	High-Level-Synthese	297
7.3.1	Die wesentlichen Schritte der High-Level-Synthese	303
7.3.2	Allokierung	304
7.3.3	Zeitablaufplanung (Scheduling)	306
7.3.4	Ressourcen-Bindung	328
7.3.5	Steuerwerksynthese	342
7.4	Register-Transfer- und Logiksynthese	345
7.4.1	Elaboration	346
7.4.2	Optimierung arithmetischer Teilnetze	348
7.4.3	Logiksynthese und Technologieabbildung	352
7.5	Zusammenfassung	359
8	Verifikation, Simulation und Test	361
8.1	Verifikation Simulation und Validierung	361
8.2	Simulation	362
8.2.1	Eine Simulationsanordnung	362
8.2.2	Simulation auf Logik- und Technologieebene	364
8.2.3	Software-Simulation	365
8.2.4	Debugging in Eingebetteten Systemen	365
8.2.5	Software-Debugging	365
8.2.6	Hardware-Debugging	366
8.3	Formale Verifikation	368
8.3.1	Aussagen-Logik	369
8.3.2	Prädikatenlogik erster Ordnung: FOL	370
8.3.3	Model Checking	370
8.3.4	Higher Order Logic	371
8.4	Werkzeuge für Modellierung und Simulation	371
8.5	Test	373
8.5.1	Begriffsbestimmungen, Ausbeute, Black-Box- und White-Box-Test	374
8.5.2	Ein klassisches Fehlermodell in der Chipproduktion	377
8.5.3	Testmuster	378
8.5.4	JTAG Boundary-Scan	378
8.6	Zusammenfassung	380
9	Mikroprozessor-Grundlagen: vom MP zum SoC	381
9.1	Evolution der Mikroprozessoren	381
9.2	Mikroprozessoren in Eingebetteten Systemen	382
9.2.1	Mikroprozessor-Architekturen	383

9.2.2	Ein- und Ausgabe durch Befehle und Interrupts	388
9.2.3	Speicher-Systeme	390
9.2.4	Wozu brauchen wir Caches?	391
9.2.5	Hauptspeicher	395
9.2.6	Festwertspeicher (ROM)	397
9.2.7	Befehls-Verarbeitungsmethoden und Pipelining	398
9.2.8	Performanz	402
9.2.9	Leistungsaufnahme-Steuerung (Power Management)	403
9.2.10	Ein-Ausgabe-Geräte und Schnittstellen	405
9.3	Mikrocontroller	407
9.3.1	Niedrigpreis-Mikrocontroller	407
9.3.2	Mikrocontroller höherer Leistung	408
9.4	Multi-Core- und Mehrprozessorsysteme	408
9.5	Mikroprozessor-Familien	410
9.6	Zusammenfassung	419
10	Kommunikation und Netzwerke	421
10.1	Das ISO-OSI-Referenzmodell	421
10.2	Der parallele Bus	424
10.2.1	Schema eines Bustreibers	425
10.2.2	Bus-Kommunikation	427
10.2.3	Bus-Zeitablaufpläne	427
10.2.4	Multiprozessor- und Multibus-Systeme	431
10.2.5	Bekannte Bus-Protokolle	433
10.2.6	AMBA-Bus-Systeme	433
10.3	Netzwerke	441
10.3.1	Kommunikationsmodi	442
10.3.2	Netzwerk-Topologien	442
10.3.3	Datenformatierung	444
10.4	Busähnliche Netzwerke oder serielle Busse	445
10.4.1	Der I^2C -Bus	447
10.4.2	Profibus (Process Field Bus)	449
10.4.3	PCI Express	451
10.4.4	InfiniBand	451
10.4.5	Ethernet	452
10.4.6	Internet	453
10.5	Bussysteme im Kraftfahrzeug	453
10.5.1	Der CAN-Bus	455
10.5.2	FlexRay, LIN und MOST	461
10.5.3	Automotive Ethernet	464
10.5.4	Zusammenfassung: Bussysteme im Kraftfahrzeug	466
10.6	Synchronisierung	467

10.6.1	Kommunikationsprimitive	467
10.6.2	Synchronisierung von Prozessorelementen	469
10.7	Sensornetzwerke	471
10.7.1	Drahtlose Sensornetzwerke	471
10.7.2	Einsatz von Sensornetzwerken, Topologie	473
10.7.3	Kommunikation im Sensornetzwerk	474
10.7.4	Der Sensorknoten	483
10.7.5	Beispiele von Sensorknoten	485
10.7.6	Kommunikationsstandards für drahtlose Netzwerke	487
10.8	Zusammenfassung	489
Literaturverzeichnis		491
Index		507