

# Inhalt

<b>1</b>	<b>Einleitung — 1</b>
1.1	Lehrbuchinhalte — 2
1.1.1	Lernziele — 4
1.1.2	Softwareentwicklungsumgebungen — 5
1.1.3	Vorausgesetzte Kenntnisse — 5
1.2	Automatisiertes und autonomes Fahren — 6
1.3	Kapitelübersicht — 7
<b>2</b>	<b>Modellbasierte Softwareentwicklung — 10</b>
2.1	Prozess der modellbasierten Softwareentwicklung — 11
2.2	Modellgetriebene Softwareentwicklung — 14
2.3	Modelle — 15
2.3.1	Modelle in der modellbasierten Softwareentwicklung — 15
2.3.2	Umgebungsmodelle für Mensch, Regelstrecke, Sensoren, Aktuatoren, Umfeld — 17
2.3.3	Modelle für Applikationssoftware des Steuergeräts — 18
2.4	Funktionsmodellierung — 19
2.5	Softwaremodellierung — 21
2.6	Validierung und Verifikation — 22
2.6.1	Model-in-the-Loop-Tests (MiL-Tests) — 23
2.6.2	Software-in-the-Loop-Tests (SiL-Tests) — 24
2.6.3	Processor-in-the-Loop-Tests (PiL-Tests) — 26
2.6.4	Hardware-in-the-Loop-Tests (HiL-Tests) — 27
2.6.5	System- und Fahrversuche — 28
<b>3</b>	<b>Laborprojekt Mini-Auto-Drive — 30</b>
3.1	Laborinhalte — 30
3.1.1	Modellbasierte und modellgetriebene Softwareentwicklung — 31
3.1.2	Installation der Entwicklungsumgebungen — 32
3.1.3	Laboraufgaben für MATLAB/Simulink oder C++ — 33
3.2	Automatisierte Fahrzeugführung — 33
3.2.1	Navigation und globale Planung — 34
3.2.2	Selbstlokalisierung — 35
3.2.3	Umfelderfassung — 35
3.2.4	Manövermanagement und lokale Planung — 35
3.2.5	Betriebsmodusmanagement und Bewegungsregelung — 37
3.2.6	Lenk-, Antriebs-, Bremsregelung, Energiemanagement — 37
3.2.7	Health-Monitoring und Safety-Management — 37
3.2.8	Fahrfunktionen in Mini-Auto-Drive (MAD) — 38

3.3	Systemübersicht — 38
3.4	Softwarearchitektur — 40
3.4.1	Reales MAD-System — 40
3.4.2	MAD-Simulation in ROS — 43
3.4.3	MAD-Simulation in MATLAB/Simulink — 44
3.5	Robot-Operating-System (ROS) — 46
3.5.1	Funktionsmerkmale — 46
3.5.2	Installation — 47
3.5.3	Hello World — 48
3.5.4	ROS-Nodes — 63

**4 Grundlagen der Signale und Systeme — 65**

4.1	Systeme — 66
4.1.1	Systemgrenze — 67
4.1.2	Hierarchisches System — 67
4.1.3	Mechatronisches System — 68
4.1.4	Kausalität — 69
4.2	Signale — 69
4.2.1	Zeitbereich — 70
4.2.2	Wertebereich — 71
4.2.3	Ortsabhängigkeit — 73
4.2.4	Kardinalität — 74
4.2.5	Informationsgehalt — 75
4.2.6	Periodizität — 75
4.2.7	Systembezug — 76
4.3	Zeitkontinuierliche, lineare und nichtlineare Modelle — 77
4.3.1	Zustandsraummodell — 78
4.3.2	Zustandsraummodell als Signalflussplan — 80
4.3.3	Beispiel: PT1-Glied — 81
4.3.4	Beispiel: PT2-Glied — 83
4.3.5	Numerik in Simulink und Boost-Odeint — 84
4.3.6	Modellierungsrichtlinien für Simulink — 85
4.3.7	Beispiel: PT1-Glied in Simulink — 87
4.3.8	Beispiel: PT2-Glied in Simulink — 96
4.3.9	Beispiel: PT1-Glied in C++ — 100
4.3.10	Beispiel: PT1-Glied in C++ und ROS — 104
4.3.11	Beispiel: PT2-Glied in C++ und ROS — 108
4.4	Zeitkontinuierliche, lineare, zeitinvariante Modelle — 111
4.4.1	Linearität — 111
4.4.2	Zeitinvarianz — 112
4.4.3	Zustandsraummodell — 113
4.4.4	Laplace-Transformation — 114

4.4.5	Übertragungsfunktion — 115
4.4.6	Systemdifferentialgleichung — 118
4.4.7	Frequenzgang — 119
4.4.8	Linearisierung — 125
4.5	Zeitdiskrete Modelle — 127
4.5.1	Abtastung im Regelkreis — 127
4.5.2	Zeitdiskretisierung — 131
4.5.3	Beispiel: Zeitdiskretisierung eines PT1-Glieds — 135
4.5.4	z-Transformation — 137
4.5.5	Übertragungsfunktion — 139
4.5.6	Diskretisierung der Übertragungsfunktion im Frequenzbereich — 141
4.5.7	Beispiel: Übertragungsfunktionen eines PT1-Glieds — 143
4.5.8	Beispiel: Zeitdiskretes PT1-Glied in Simulink — 145
4.5.9	Modellierungsrichtlinien für Simulink — 149
<b>5</b>	<b>Fahrdynamiksimulation — 150</b>
5.1	Longitudinaldynamikmodell — 150
5.1.1	Signalflussplan — 151
5.1.2	Ersatzschaltbilder — 152
5.1.3	Zustandsraummodell — 154
5.2	Kinematisches Einspurmodell für Hinterachsmittpunkt — 155
5.2.1	Zustandsraummodell — 157
5.2.2	Signalflussplan — 158
5.2.3	Modellherleitung — 158
5.3	Kinematisches Einspurmodell für Vorderachsmittpunkt — 160
5.3.1	Zustandsraummodell — 160
5.3.2	Modellherleitung — 161
5.4	Kinematisches Einspurmodell für Longitudinalachsenpunkt — 162
5.4.1	Zustandsraummodell — 162
5.4.2	Modellherleitung — 163
5.4.3	Erweiterung des Kinematikmodells um die Longitudinaldynamik — 163
5.5	Dynamisches Einspurmodell mit Reifenkräften — 167
5.5.1	Kinematik des Schwerpunkts — 168
5.5.2	Dynamik des Schwerpunkts — 169
5.5.3	Schräglauwinkel der Räder — 171
5.5.4	Pacejka's Magic-Formula-Modell für Reifenkräfte — 171
5.5.5	Kombination des Einspurmodells und des Longitudinaldynamikmodells — 172
5.6	Aufgaben — 176
5.6.1	Aufgabe 5.1 Longitudinaldynamikmodell [C++/Simulink] — 176
5.6.2	Aufgabe 5.2 Fahrdynamiksimulation [Simulink] — 177

5.6.3	Aufgabe 5.3 Fahrdynamiksimulation [C++] — 178
<b>6</b>	<b>Geschwindigkeitsregelung — 182</b>
6.1	PI-Reglerentwurf für PT1-Streckendynamik — 183
6.1.1	Modell der Regelstrecke — 184
6.1.2	Entwurf des PI-Reglers — 185
6.1.3	Regelkreisanalyse — 187
6.2	Stabilität und Robustheit eines Regelkreises — 190
6.2.1	Charakteristische Gleichungen — 190
6.2.2	Vereinfachtes Nyquist-Kriterium — 192
6.2.3	Beispiel: I-Regelung einer PT2-Strecke — 193
6.2.4	Amplituden- und Phasenränder — 196
6.2.5	Verallgemeinertes Nyquist-Kriterium — 198
6.3	Geschwindigkeitsregelung für totzeitbehaftete PT1-Strecke — 199
6.3.1	Anforderungen an Führungsgröße — 199
6.3.2	Anforderungen an Regelgröße — 200
6.3.3	Anforderungen an Stellgröße — 200
6.3.4	Anforderungen an Regelkreisdynamik — 201
6.4	Aufgaben — 201
6.4.1	Aufgabe 6.1 Entwurf des Geschwindigkeitsreglers [C++/Simulink] — 201
6.4.2	Aufgabe 6.2 Simulink-Subsystem für Geschwindigkeitsregelung [Simulink] — 202
6.4.3	Aufgabe 6.3 ROS-Node carctrl_node für Geschwindigkeitsregelung [C++] — 205
<b>7</b>	<b>Longitudinalpositionsregelung — 208</b>
7.1	Kaskadenregelung — 208
7.1.1	Innerer Geschwindigkeitsregelkreis — 209
7.1.2	Äußerer Positionsregelkreis — 210
7.1.3	Reglerentwurf — 211
7.2	Vorsteuerung — 212
7.2.1	Entwurf der Vorsteuerung durch Invertierung der Regelstrecke — 213
7.2.2	Differenzgrad der Regelstrecke — 213
7.3	Positionsregelung mit Kaskadenregelung und Vorsteuerung — 215
7.3.1	Entwurf der Vorsteuerung — 216
7.3.2	Entwurf der Führungssignalgenerierung — 217
7.4	Aufgaben — 221
7.4.1	Aufgabe 7.1 Entwurf der Longitudinalpositionsregelung [C++/Simulink] — 221
7.4.2	Aufgabe 7.2 Erweiterung des Simulink-Subsystems Control Software für Longitudinalpositionsregelung [Simulink] — 223

7.4.3	Aufgabe 7.3 Erweiterung des ROS-Nodes carctrl_node für Longitudinalpositionsregelung [C++] — 225
<b>8</b>	<b>Bahnkurvendefinition — 227</b>
8.1	Sollbahnkurven — 227
8.1.1	Abgeleitete Bahngrößen — 228
8.1.2	Kreisbögen — 230
8.1.3	Klothoide — 233
8.1.4	MAD-Library in C++ — 237
8.1.5	MODBAS-CAR-Library in MATLAB — 239
8.2	Interpolation mit kubischen Splines — 240
8.2.1	Abschnittsweise definierte kubische Polynome — 241
8.2.2	Berechnung von kubischen Splines in MATLAB — 245
8.2.3	Ableitungen kubischer Splines — 247
8.3	Aufgaben — 248
8.3.1	Aufgabe 8.1 Gerade Bahnkurve [Simulink/C++] — 248
8.3.2	Aufgabe 8.2 MODBAS-CAR-Funktionen für Klothoide [Simulink] — 248
8.3.3	Aufgabe 8.3 ROS-Node track_node mit Kreisverkehr [C++] — 250
<b>9</b>	<b>Bahnfolgeregelung — 252</b>
9.1	Führungssignalgenerierung — 253
9.1.1	MAD-Library in C++ — 255
9.1.2	MODBAS-CAR-Library in MATLAB — 255
9.2	Dynamik der Regelabweichung — 256
9.2.1	Nichtlineare Fehlerdynamik — 257
9.2.2	Linearisierte Fehlerdynamik — 259
9.3	Zustandsregler — 260
9.4	Steuerbarkeit — 262
9.5	Nichtlineare Vorsteuerung — 264
9.6	Aufgaben — 265
9.6.1	Aufgabe 9.1 Simulink-Subsystem Control Software für Geschwindigkeits- und Bahnfolgeregelung [Simulink] — 265
9.6.2	Aufgabe 9.2 ROS-Node carctrl_node für Geschwindigkeits- und Bahnfolgeregelung [C++] — 267
9.6.3	Aufgabe 9.3 Mini-Auto-Drive-Wettbewerb [Simulink / C++] — 269
<b>Literaturverzeichnis — 272</b>	
<b>Register — 274</b>	