

Inhalt

Vorwort.....	V
1 Anwendung von Bussystemen und Protokollen	1
1.1 Überblick	2
2 Kfz-Bussysteme, Protokolle und Standards	5
2.1 Standardisierung bei Bussystemen	7
3 Kfz-Bussysteme – Physical und Data Link Layer.....	9
3.1 Grundlagen für Kfz-Bussysteme	9
3.1.1 Elektrotechnische Grundlagen.....	9
3.1.2 Topologie und Kopplung von Bussystemen	12
3.1.3 Botschaften, Protokollstapel, Dienste (Services).....	14
3.1.4 Kommunikationsmodelle, Adressierung	16
3.1.5 Zeichen- und Bitstrom-basierte Übertragung, Nutzdatenrate.....	20
3.1.6 Buszugriffsverfahren, Fehlererkennung und Fehlerkorrektur.....	22
3.1.7 Jitter und Latenz bei der Datenübertragung	25
3.2 K-Line nach ISO 9141 und ISO 14230	26
3.2.1 Entwicklung von K-Line und KWP 2000	26
3.2.2 K-Line Bus-Topologie und Physical Layer.....	27
3.2.3 Data Link Layer.....	29
3.2.4 Einschränkungen für emissionsrelevante Komponenten (OBD).....	33
3.2.5 Schnittstelle zwischen Protokoll-Software und Kommunikations- Controller	33
3.2.6 Ältere K-Line-Varianten	34
3.2.7 Zusammenfassung K-Line – Layer 1 und 2.....	35
3.3 Controller Area Network CAN nach ISO 11898	36
3.3.1 Entwicklung von CAN	36
3.3.2 Bus-Topologie und Physical Layer	37
3.3.3 CAN Data Link Layer	39
3.3.4 Fehlerbehandlung	41
3.3.5 Einsatz von CAN – Höhere Protokolle.....	42
3.3.6 Schnittstelle zwischen Protokoll-Software und CAN-Controller	43
3.3.7 Time-Triggered-CAN (TTCAN) – Deterministischer Buszugriff.....	46
3.3.8 Zusammenfassung CAN – Layer 1 und 2	48
3.4 Local Interconnect Network LIN.....	49
3.4.1 Überblick.....	49
3.4.2 Data Link Layer.....	50
3.4.3 Zeitsynchrones Senden von Botschaften (Message Scheduling).....	53
3.4.4 Neue Botschaftstypen bei LIN V2.0.....	53
3.4.5 LIN Transportschicht und ISO Diagnose über LIN	54
3.4.6 LIN Configuration Language.....	56

3.4.7	Dynamische Konfiguration von LIN-Slave-Steuergeräten	59
3.4.8	LIN Application Programming Interface (API).....	61
3.4.9	Zusammenfassung LIN – Layer 1 und 2	63
3.5	FlexRay	64
3.5.1	Bus-Topologie und Physical Layer	64
3.5.2	Data Link Layer.....	66
3.5.3	Netzwerk-Start und Takt-Synchronisation	70
3.5.4	Fehlerbehandlung, Bus Guardian.....	72
3.5.5	Konfiguration und übergeordnete Protokolle	73
3.5.6	Beispiel-Konfiguration – Schnittstelle zum FlexRay-Controller	74
3.5.7	Weiterentwicklung.....	79
3.5.8	Zusammenfassung FlexRay – Layer 1 und 2.....	80
3.6	SAE J1850	81
3.7	Media Oriented Systems Transport MOST	84
3.7.1	Bus-Topologie und Physical Layer	84
3.7.2	Data Link Layer.....	85
3.7.3	Kommunikationscontroller	90
3.7.4	Network Services und Funktionsblöcke.....	91
3.7.5	Netzmanagement.....	95
3.7.6	Höhere Protokollschichten.....	97
3.7.7	Beispiel für Systemstart und Audioverbindung.....	97
3.7.8	Neuere Entwicklungen: MOST150	99
3.7.9	Zusammenfassung MOST	101
3.8	Sensor-Aktor-Bussysteme	102
3.8.1	SENT – Single Edge Nibble Transmission nach SAE J2716	102
3.8.2	PSI 5 – Peripheral Sensor Interface 5	103
3.8.3	ASRB 2.0 – Automotive Safety Restraint Bus (ISO 22898)	104
3.9	Normen und Standards	107
4	Transportprotokolle	109
4.1	Transportprotokoll ISO TP für CAN nach ISO 15765-2	109
4.1.1	Botschaftsaufbau	110
4.1.2	Flusssteuerung, Zeitüberwachung und Fehlerbehandlung	111
4.1.3	Dienste für die Anwendungsschicht (Application Layer Services)	113
4.1.4	Protokoll-Erweiterungen	114
4.1.5	Adressierung bei KWP 2000/UDS – Zuordnung von CAN Identifiern	114
4.2	Transportprotokoll AUTOSAR TP für FlexRay.....	115
4.3	Transportprotokoll TP 2.0 für CAN.....	118
4.3.1	Adressierung und CAN Message Identifier	118
4.3.2	Broadcast-Botschaften	119
4.3.3	Dynamischer Kanalaufbau und Verbindungsmanagement	120
4.3.4	Datenübertragung	122
4.4	Transportprotokoll TP 1.6 für CAN.....	124
4.4.1	Botschaftsaufbau	124
4.4.2	Dynamischer Kanalaufbau.....	124
4.4.3	Datenübertragung und Datenrichtungswechsel	125

4.5	Transportprotokoll SAE J1939/21 für CAN	126
4.5.1	Übertragungsarten, Adressierung und CAN Message Identifier	126
4.5.2	Segmentierte Datenübertragung (Multi Packet)	129
4.6	Normen und Standards	132
5	Diagnoseprotokolle – Application Layer	133
5.1	Diagnoseprotokoll KWP 2000 (ISO 14230-3)	135
5.1.1	Überblick	135
5.1.2	Diagnosesitzungen (Diagnostic Management)	138
5.1.3	Adressierung der Steuergeräte nach KWP 2000 und UDS	140
5.1.4	Bussystem-abhängige Dienste (Network Layer Protocol Control)	142
5.1.5	Fehlerspeicher lesen und löschen (Stored Data Transmission)	143
5.1.6	Daten lesen und schreiben (Data Transmission), Ansteuern von Steuergeräte-Ein- und Ausgängen (Input/Output Control)	144
5.1.7	Speicherblöcke auslesen und speichern (Upload, Download)	145
5.1.8	Start von Programmen im Steuergerät (Remote Routine Activation)	146
5.1.9	Erweiterte Dienste (Extended Services)	147
5.2	Unified Diagnostic Services UDS nach ISO 14229/15765-3	147
5.2.1	Unterschiede zum KWP 2000 Diagnoseprotokoll	147
5.2.2	Überblick über die UDS-Diagnosedienste	148
5.2.3	Response on Event-Dienst	155
5.3	On-Board-Diagnose OBD nach ISO 15031 / SAE J1979	157
5.3.1	Überblick OBD-Diagnosedienste	157
5.3.2	Auslesen des Fehlerspeichers und von Steuergerätewerten	159
5.3.3	Abfrage der Testergebnisse für abgasrelevante Komponenten	162
5.3.4	OBD-Fehlercodes	163
5.3.5	Data Link Security	165
5.3.6	Pass-Through-Programmierung	165
5.3.7	Beispiel	166
5.4	Normen und Standards	168
6	Anwendungen für Messen, Kalibrieren und Diagnose (ASAM AE MCD)	169
6.1	Einführung	169
6.2	Busprotokolle für Aufgaben in der Applikation (ASAM AE MCD 1MC)	173
6.2.1	CAN Calibration Protocol CCP	174
6.2.2	Extended Calibration Protocol XCP	182
6.2.3	AML-Konfigurationsdateien für XCP und CCP	197
6.2.4	Interface zwischen Busprotokolltreiber und Applikationssystem ASAM MCD 1b	199
6.3	Field Bus Exchange Format FIBEX	203
6.4	Überblick über ASAM AE MCD 2 und MCD 3	212
6.5	Applikationsdatensätze nach ASAM MCD 2 MC	213
6.5.1	ASAP2/A2L-Applikationsdatensätze	213
6.5.2	Calibration Data Format CDF, Meta Data Exchange Format MDX	217

6.6 ODX-Diagnosedatensätze nach ASAM AE MCD 2D.....	219
6.6.1 Aufbau des ODX-Datenmodells.....	220
6.6.2 DIAG-LAYER: Hierarchische Diagnosebeschreibung.....	222
6.6.3 VEHICLE-INFO-SPEC: Fahrzeugzugang und Bustopologie	225
6.6.4 COMPARAM-SPEC und COMPARAM-SUBSET: Busprotokoll	228
6.6.5 DIAG-COMM und DIAG-SERVICE: Diagnosedienste.....	230
6.6.6 Einfache und komplexe Datenobjekte	235
6.6.7 SINGLE-ECU-JOB und MULTIPLE-ECU-JOB: Diagnoseabläufe.....	243
6.6.8 STATE-CHART: Diagnosesitzungen	245
6.6.9 ECU-CONFIG: Beschreibung der Steuergeräte-Konfiguration	246
6.6.10 ECU-MEM: Beschreibung der Flash-Programmierung	247
6.6.11 FUNCTION-DICTIONARY: Funktionsorientierte Diagnose.....	249
6.6.12 Packed ODX und ODX-Autorenwerkzeuge	250
6.6.13 ODX Version 2.2.....	250
6.7 ASAM AE MCD 3 – Server	251
6.7.1 Funktionsgruppe M – Messen	252
6.7.2 Funktionsgruppe C – Kalibrieren	254
6.7.3 Funktionsgruppe D – Diagnose	254
6.8 MVCI-Schnittstelle für Diagnosetester nach ISO 22900.....	256
6.9 Normen und Standards.....	259
7 Software-Standards: OSEK, AUTOSAR, HIS.....	261
7.1 Einführung.....	261
7.2 OSEK/VDX.....	264
7.2.1 Ereignisgesteuerter Betriebssystemkern OSEK/VDX OS	265
7.2.2 Kommunikation in OSEK/VDX COM	275
7.2.3 Netzmanagement mit OSEK/VDX NM.....	279
7.2.4 Zeitgesteuerter Betriebssystemkern OSEK Time und fehlertolerante Kommunikation OSEK FTCOM.....	284
7.2.5 Erweiterung von OSEK OS um Schutzmechanismen: Protected OS	287
7.3 Hardware-Ein- und Ausgabe (HIS IO Library, IO Driver).....	287
7.4 HIS Hardwaretreiber für CAN-Kommunikationscontroller (HIS CAN Driver).....	289
7.5 HIS Flash-Lader	289
7.6 AUTOSAR.....	290
7.6.1 Überblick über die AUTOSAR-Basissoftware.....	292
7.6.2 Betriebssystem AUTOSAR OS.....	302
7.6.3 Kommunikationsstack AUTOSAR COM und Diagnose DCM	306
7.6.4 Netzmanagement AUTOSAR NM	316
7.6.5 Virtual Function Bus VFB, Runtime Environment AUTOSAR RTE und AUTOSAR Softwarekomponenten	320
7.6.6 Ausblick.....	325
7.7 Normen und Standards.....	327

8 Werkzeuge, Anwendungen und Einsatzgebiete	329
8.1 Softwarekomponenten für Steuergeräte	329
8.2 Entwurf und Test der On-Board-Kommunikation	329
8.2.1 Entwicklungsprozess mit <i>CANoe</i> von Vector Informatik	330
8.2.2 Netzwerkdesign mit dem <i>DaVinci Network Designer</i>	330
8.2.3 Simulation des Gesamtsystems in <i>CANoe</i>	334
8.2.4 Restbussimulation als Entwicklungsumgebung für Steuergeräte	335
8.2.5 Integration des Gesamtsystems	337
8.3 Werkzeuge zur Applikation von Steuergeräten	338
8.4 Flash-Programmierung von Steuergeräten	340
8.4.1 Rahmenbedingungen	341
8.4.2 Flash-Speicher	344
8.4.3 Flash-Programmierprozess	346
8.4.4 Beispiel eines Flash-Laders: <i>ADLATUS</i> von SMART Electronic	354
8.4.5 Test und Freigabe von Flash-Ladern und Busprotokollen	360
8.5 Diagnosewerkzeuge in Entwicklung und Fertigung	364
8.5.1 Beispiel für Diagnosewerkzeuge: <i>samDia</i> von Samtec Automotive	365
8.6 Autorenwerkzeuge für Diagnosedaten	373
8.6.1 Beispiel für Autorenwerkzeuge: <i>ODXplorer</i> von Samtec Automotive	375
8.7 ASAM MCD3-Laufzeitsysteme	380
8.7.1 Beispiel für ein Diagnosesystem: <i>samMCD3Server</i> von Samtec Automotive Software & Electronics	381
9 Kommunikation zwischen Fahrzeugen	387
9.1 Mautsysteme	387
9.2 Car2Car-Konsortium	388
9.3 Normen und Standards	390
Literaturverzeichnis	391
Web-Adressen	392
Abkürzungen	394
Sachwortverzeichnis	400