

Inhaltsverzeichnis

1	Elektronik im Fahrzeug	1
1.1	Historische Entwicklung der Elektronik im Fahrzeug	1
1.2	Steuergeräte-Hardware	4
1.3	Steuergeräte-Software	7
1.4	Standardisierung	7
1.5	Zusammenfassung und Ausblick	9
2	Grundlagen der Datenkommunikation	11
2.1	Kommunikation	11
2.2	Kommunikationsarten	12
2.2.1	Nicht bestätigte Kommunikation	12
2.2.2	Bestätigte Kommunikation (Client-Server-Modell)	14
2.2.3	Indirekte Kommunikation über einen Verbindungsknoten (Gateway)	15
2.2.4	Bestätigte Broadcast-Kommunikation	16
2.3	Das generische ISO/OSI-Schichtenmodell im Überblick	18
2.3.1	Ableitung der Diagnose-Terminologie	21
2.3.2	Die ISO/OSI-Header	22
2.3.3	Protocol Data Unit – PDU	23
2.3.4	Diagnosedienste und das ISO/OSI-Modell	24
2.4	Vernetzungstopologien	25
3	Bussysteme	29
3.1	Historische Entwicklung	29
3.2	SAE J1850	31
3.3	K-Leitung	31
3.3.1	Bus-Physik (Physical Layer)	33
3.3.2	Aufbau einer Nachricht auf der K-Leitung nach ISO 14230	33
3.3.3	Initialisierung der Kommunikation	35
3.4	CAN	37
3.4.1	Historische Entwicklung	37
3.4.2	CAN-Bus Übersicht	38
3.4.3	CAN-Controller	40
3.4.4	CAN Physical Layer	41
3.4.5	CAN-Frames (Data Link Layer)	44
3.4.6	Fehlermanagement (-erkennung)	47

3.5	LIN 2.1	50
3.5.1	Historische Entwicklung	50
3.5.2	Übersicht	51
3.5.3	Physical Layer Specification	52
3.5.4	Protocol Specification	53
3.5.5	Diagnostic and Configuration Specification	57
3.6	FlexRay 2.1	59
3.6.1	Topologien	61
3.6.2	Physical Layer	63
3.6.3	Data Link Layer	64
3.6.4	Uhrensynchronisation	68
3.7	Multimedia-Netzwerke	69
3.7.1	Einleitung	69
3.7.2	MOST [®]	69
3.7.3	IDB 1394	73
3.8	Vernetzte Netzwerke	74
4	Protokolle	77
4.1	Einführung in die Diagnosekommunikation	77
4.1.1	Onboard-Kommunikation	78
4.1.2	Offboard-Kommunikation	79
4.1.3	Zusammenfassung der Steuergerätefunktionen	81
4.2	Diagnoseprotokolle	82
4.3	KWP 2000	84
4.4	Diagnostics on CAN	85
4.5	Adaption des ISO/OSI-Modells für die Diagnosekommunikation	86
4.6	Prinzipieller Aufbau von Diagnosebotschaften	88
4.6.1	Diagnosebotschaft vom Typ Request	89
4.6.2	Diagnosebotschaft vom Typ Response	89
4.6.3	Diagnosebotschaft vom Typ Positive Response	89
4.6.4	Diagnosebotschaft vom Typ Negative Response	89
4.6.5	Diagnosebotschaft vom Typ „Keine Antwort“	90
4.6.6	Service-Parameter	90
4.6.7	Zusammenfassung und Beispiel	95
4.7	Functional Units	96
4.8	Techniken der Diagnoseprotokolle	96
4.8.1	Adressvergabe	96
4.8.2	Abläufe in der Diagnosekommunikation	102
4.8.3	Adaption der Diagnosebotschaft in die Network und Transport Layer	106
4.8.4	Segmentierung	108
5	Unified Diagnostic Services	119
5.1	Die normierten Services	122
5.1.1	Diagnose- und Kommunikationsmanagement	124
5.1.2	Datenübertragung	125
5.1.3	Datenübertragung gespeicherter Daten	127

5.1.4	Ein- und Ausgabesteuerung	127
5.1.5	Fernaktivierung von Funktionen	127
5.1.6	Übertragung von Speicherinhalten des Steuergerätes	127
5.2	Aufbau der UDS-Botschaften	128
5.2.1	Das Sub-Function Byte	130
5.2.2	Hinweise zum Response Handling in UDS	133
5.2.3	Zusammenfassung der UDS-Botschaften	135
5.3	UDS-Kommunikationsabläufe	136
5.3.1	Einfacher Ablauf ohne Sub-Function Byte	136
5.3.2	Einfacher Ablauf mit Sub-Function Byte	137
5.3.3	Periodischer Ablauf	138
5.3.4	Ereignisgesteuerter Ablauf	144
5.4	Diagnose-Sessions	154
5.4.1	UDS Sessions	154
5.4.2	Session-Wechsel	155
5.4.3	Rücksetzen der Zustandseigenschaften	156
5.5	Beispiel einer Diagnosesitzung	158
5.6	Schlussbetrachtung des UDS	163
5.6.1	Unterschiede KWP 2000 und UDS	163
5.6.2	Allgemeine Hinweise zu UDS	164
6	Kommunikationssysteme	167
6.1	Übersicht	167
6.2	Proprietäre Kommunikationssysteme	169
6.2.1	EDIS	169
6.2.2	E-Tester	170
6.3	ASAM MCD und ISO MVCI	172
6.3.1	Der ASAM e.V.	172
6.3.2	ASAM AE	173
6.4	D-Server	175
6.4.1	Interner Aufbau eines D-Servers	176
6.4.2	Umsetzungsbeispiel Diagnostic Tool Set	178
6.4.3	Kernaufgabe eines D-Servers	179
6.4.4	Die Programmierschnittstelle des D-Servers	180
6.4.5	Generischer Ablauf einer D-Applikation	181
6.4.6	Job-Ausführung auf dem D-Server	182
6.4.7	Job-Aufbau und Erzeugung	184
6.5	Schnittstellen zur VCI-Hardware	188
6.5.1	Proprietäre Lösungen zur Ankopplung von VCI-Hardware	188
6.5.2	Der Communication Processor und die D-PDU API	190
6.5.3	Migrationsszenarien	191
6.5.4	Kostenreduktion durch den Einsatz der D-PDU API	192
6.6	Interface-Hardware	192
6.6.1	CAN-Interfaces	193
6.6.2	Diagnose-Interfaces	194

7 Datenbasen	199
7.1 Einleitung	199
7.2 Der Standard ODX	200
7.2.1 Was ist ODX?	200
7.2.2 Diagnosedaten	201
7.2.3 Austauschprozesse	204
7.2.4 Entstehung einer Diagnosespezifikation	206
7.2.5 Historische Entwicklung des ODX-Standards	209
7.3 Aufbau des Standards ODX V2.0.1	210
7.3.1 Die fünf ODX-Kategorien	211
7.3.2 Diag-Layer und Diag-Layer-Container	213
7.3.3 Spezifikation der Kommunikationsparameter	215
7.3.4 Fahrzeuginformationen und -topologie	217
7.3.5 Programmierbeschreibungen	220
7.3.6 Steuergeräteübergreifende Jobs	221
7.4 Datenbasis vs. Data-Pool	222
7.4.1 Die ODX-Datenbasis	222
7.4.2 ODX-Data-Pool	223
7.4.3 Redundanzvermeidung	224
7.5 Elemente und Attribute des ODX	228
7.5.1 Aufbau des Diagnosedienstes in ODX	229
7.5.2 Simple Data Object Property	232
7.5.3 Computational Methods	237
7.5.4 Zusammenfassung der Datenprozessierung	241
7.5.5 Komplexe und erweiterte Datentypen	242
7.5.6 Complex Data Object Properties	243
7.5.7 Tabellen	245
7.5.8 DTC-Beschreibungsobjekte	247
7.6 Vererbung	253
7.6.1 Import vs. Vererbung	254
7.6.2 Fallbeispiele für Vererbungsbeziehungen	256
7.7 Autorensysteme	257
7.7.1 Austauschformat XML	258
7.7.2 Unterstützung durch das Autorensystem	260
7.7.3 Herstellung eines MCD-Projektes	262
7.7.4 Datenqualität	262
7.7.5 Import und Export von Daten	264
7.8 Packaged ODX (PDX)	264
7.8.1 Eigenschaften des PDX	264
7.9 Neuerungen mit ODX 2.x (ISO 22901)	265
7.9.1 Anzahl der ODX-Kategorien	266
7.9.2 Beziehungen zwischen den Diag-Layern	270
7.9.3 Wiederverwendung kompletter DTC-DOPs	272
7.9.4 Weitere Neuheiten	273

8 Anwendungen der Datenkommunikation	275
8.1 Übersicht	275
8.2 Protokollentwicklung und -test	275
8.2.1 Eigenschaften des Werkzeuges Monaco	276
8.2.2 Die Applikation Monaco auf dem D-Server	276
8.2.3 Analyse des OBD-Dienstes Mode \$09	277
8.3 Restbussimulation	283
8.4 Gatewaytester	285
8.5 ECU-Hardware-Tester	287
8.6 Hardware In The Loop (HIL)	288
8.6.1 Anwendungsbeispiel: AutomationDesk	289
8.6.2 Anwendungsbeispiel: LABCAR	290
8.7 VAS 5163	292
8.8 Re-Programmierung und der ODX Standard	294
8.8.1 Erst-Programmierung vs. Re-Programmierung	294
8.8.2 Programmierung durch Fahrzeughersteller und Steuergerätezulieferer ..	295
8.8.3 Erst-Programmierung	295
8.8.4 Re-Programmierung	296
8.8.5 Programmierprozesse des Steuergeräteherstellers	296
8.8.6 Mechanismen des ODX Standards für die Flashprogrammierung	296
8.8.7 ODX-Idents	302
8.8.8 Speicherung der Programmierlogik und Ausführung von Sessions	305
8.8.9 Assistenten zur Befüllung der ODX-Daten	308
8.8.10 Zusammenfassung	310
8.9 Prüfsysteme in der Produktion	310
8.9.1 Prüfsystem für Lenkwinkelsensoren	310
8.9.2 EOL-Prüfsysteme	312
8.10 Flashen in der Fahrzeugmontage	316
8.11 Fahrversuche	318
8.12 Diagnose in der Werkstatt	320
8.12.1 Volkswagen-Diagnosegeräte	320
8.12.2 DAF DAVIE XDc	321
8.12.3 Elektronisches Service Tool (EST)	322
8.13 OBD-Analyse mit PDA	324
Glossar	327
Abkürzungsverzeichnis	345
Anhang	351
Weiterführende Literatur	351
SAE Standards (www.sae.org)	352
ISO-Standards (www.iso.org)	352
Sachwörterverzeichnis	355