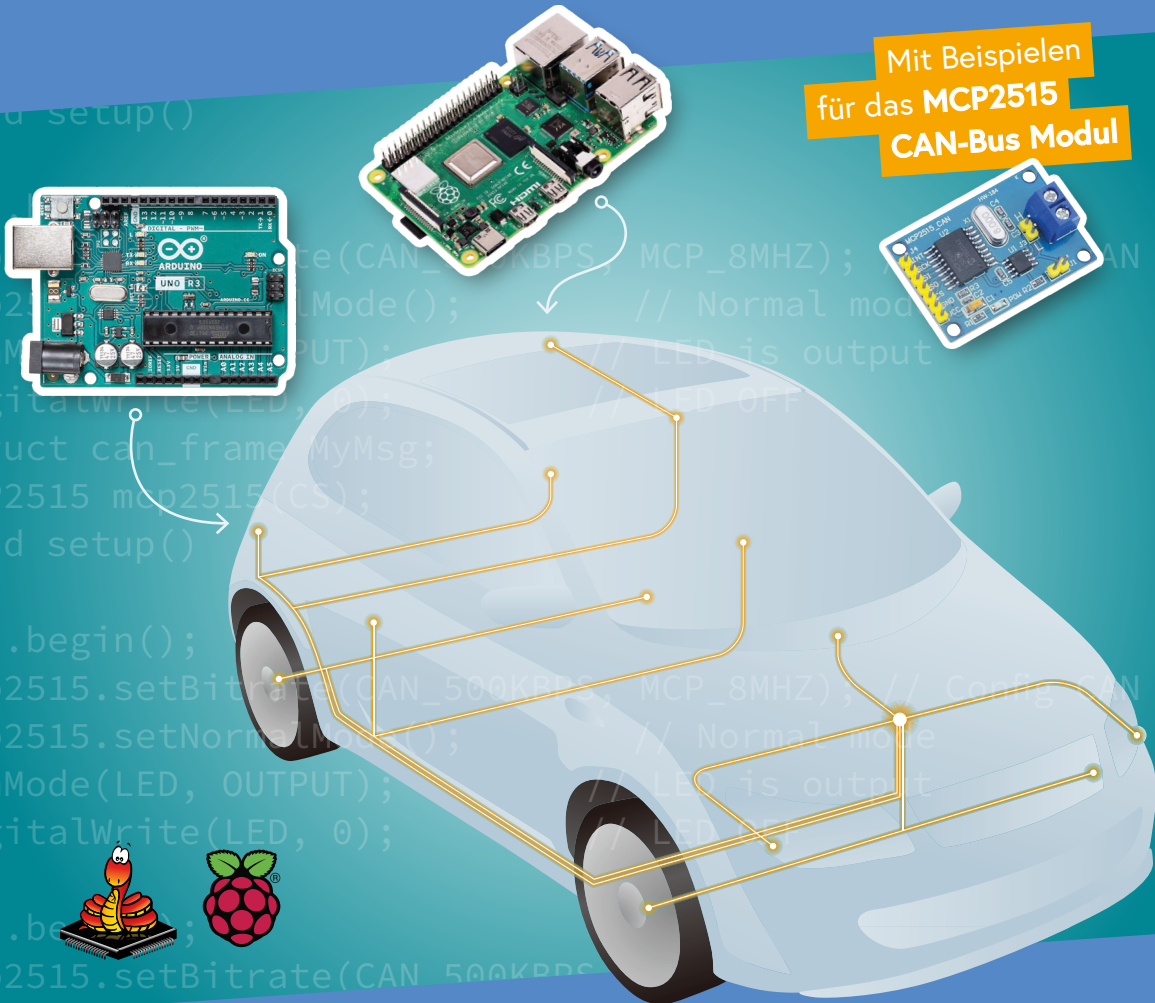


Das CAN-Bus Praxisbuch

Projekte mit Arduino Uno und Raspberry Pi



Prof. Dogan Ibrahim
Ahmet Ibrahim

Das CAN-Bus Praxisbuch

Projekte mit Arduino Uno und Raspberry Pi



Prof. Dogan Ibrahim
&
Ahmet Ibrahim

● © 2023: Elektor Verlag GmbH, Aachen.

1. Auflage 2023

● Alle Rechte vorbehalten.

Die in diesem Buch veröffentlichten Beiträge, insbesondere alle Aufsätze und Artikel sowie alle Entwürfe, Pläne, Zeichnungen und Illustrationen sind urheberrechtlich geschützt. Ihre auch auszugsweise Vervielfältigung und Verbreitung ist grundsätzlich nur mit vorheriger schriftlicher Zustimmung des Herausgebers gestattet.

Die Informationen im vorliegenden Buch werden ohne Rücksicht auf einen eventuellen Patentschutz veröffentlicht. Die in diesem Buch erwähnten Soft- und Hardwarebezeichnungen können auch dann eingetragene Warenzeichen sein, wenn darauf nicht besonders hingewiesen wird. Sie gehören dem jeweiligen Warenzeicheninhaber und unterliegen gesetzlichen Bestimmungen.

Bei der Zusammenstellung von Texten und Abbildungen wurde mit größter Sorgfalt vorgegangen. Trotzdem können Fehler nicht vollständig ausgeschlossen werden. Verlag, Herausgeber und Autor können für fehlerhafte Angaben und deren Folgen weder eine juristische Verantwortung noch irgendeine Haftung übernehmen.

Für die Mitteilung eventueller Fehler sind Verlag und Autor dankbar.

● Erklärung

Autor, Übersetzer und Verlag haben sich nach besten Kräften bemüht, die Richtigkeit der in diesem Buch enthaltenen Informationen zu gewährleisten. Sie übernehmen keine Haftung für Verluste oder Schäden, die durch Fehler oder Auslassungen in diesem Buch verursacht werden, unabhängig davon, ob diese Fehler oder Auslassungen auf Fahrlässigkeit, Versehen oder eine andere Ursache zurückzuführen sind, und lehnen jegliche Haftung hiermit ab.

● **ISBN 978-3-89576-571-1** Print
ISBN 978-3-89576-572-8 eBook

● Satz und Aufmachung: D-Vision, Julian van den Berg | Oss (NL)
Druck: Ipskamp Printing, Enschede (NL)

Elektor Verlag GmbH, Aachen
www.elektor.de

Elektor ist die weltweit wichtigste Quelle für technische Informationen und Elektronik-Produkte für Maker, Ingenieure und Elektronik-Entwickler und für Firmen, die diese Fachleute beschäftigen. Das internationale Team von Elektor entwickelt Tag für Tag hochwertige Inhalte für Entwickler und DIY-Elektroniker, die über verschiedene Medien (Magazine, Videos, digitale Medien sowie Social Media) in zahlreichen Sprachen verbreitet werden. **www.elektor.de**

Inhaltsverzeichnis

Vorwort.	9
Kapitel 1 • Kraftfahrzeug-Bussysteme.	10
1.1 Übersicht	10
1.2 Netzwerksysteme in Fahrzeugen	12
1.2.1 LIN-Bus	13
1.2.2 FlexRay	13
1.2.3 MOST	13
1.2.4 Byteflight	14
1.2.5 Intellibus	14
1.2.6 CAN-Bus	14
1.2.7 Weitere Busstandards	16
1.3 Vergleich der Bussysteme in Fahrzeugen	16
1.4 Grundstruktur eines CAN-Bus in Fahrzeugen	19
1.5 Vorteile des CAN-Busses.	22
1.6 Nachteile des CAN-Busses	23
1.7 Haupteigenschaften des CAN-Busses	23
1.8 CANopen	23
Kapitel 2 • Physische Schichtstruktur von CAN	25
2.1 Übersicht	25
2.2 Abschluss des des CAN-Busses	25
2.3 Datenübertragungsrate auf dem CAN-Bus	28
2.4 Länge von Stichleitungen	29
2.5 CAN-Bus Knoten	29
2.6 CAN-Bus Signalpegel	30
2.6.1 Spannungspegel CAN_H	31
2.6.2 Spannungspegel CAN_L	31
2.6.3 Kurvenform des CAN-Signals	31
2.6.4 Bus-Arbitrierung	32
2.6.5 Bus-Transceiver	32
2.7 CAN-Stecker	33

2.8 CAN-Repeater	36
2.9 CAN zu PC-Schnittstelle	36
Kapitel 3 • CAN-Bus Frames.	37
3.1 Data Frame.	39
3.1.1 Start of Frame (SOF).	40
3.1.2 Arbitrierungsfeld.	40
3.1.3 RTR-Feld.	42
3.1.4 Kontrollfeld	42
3.1.5 Datenfeld.	43
3.1.6 CRC-Feld	43
3.1.7 ACK-Feld	43
3.1.8 End-Of Frame-Feld	44
3.2 Remote Frame.	44
3.3 Error Frame	45
3.4 Overload Frame.	47
3.5 Erweiterte CAN-Frames	48
Kapitel 4 • Datenaustausch auf dem CAN-Bus.	50
4.1 Übersicht	50
4.2 Datenaustausch mit Data Frames	50
4.3 Remote Frames auf dem Bus	53
Kapitel 5 • CAN-Bus Schnittstellenmodul	55
5.1 Übersicht	55
5.2 MCP2515 Schnittstellenmodul für den CAN-Bus.	55
5.2.1 Das MCP2515 CAN-Controller-Chip	57
5.2.2 Das TJA1050 CAN-Transceiverchip	60
Kapitel 6 • CAN-Bus Projekte mit Arduino Uno	61
6.1 Übersicht	61
6.2 CAN-Bus Schnittstelle Arduino Uno	61
6.3 Projekt 1: Einfache CAN-Bus Kommunikation von Arduino zu Arduino	61
6.4 Projekt 2: Drucktaste und LED	70
6.5 Projekt 3: Drucktaste und LED mit CAN-Bus Interrupts.	74

6.6 Projekt 4: Temperatur-Fernalarm	77
6.7 Projekt 5: Temperatur-Abfrage	81
6.8 Projekt 6: Temperaturabfrage mit einer Drucktaste	85
6.9 Projekt 7: RGB-Darstellung mit Tasten	88
6.10 Projekt 8: Anzeige von Umgebungstemperatur und Luftfeuchte auf LCD.	93
6.11 Projekt 9: CAN-Bus mit drei Knoten: Messung von Außen- und Innentemperatur	104
6.12 Verwendung von Akzeptanzmasken und -filtern.	109
6.13 Projekt 10: CAN-Bus mit drei Knoten: Messung der Außen- und Innentemperatur mit Drucktasten	111
Kapitel 7 • Fehlerbedingungen auf dem CAN-Bus	120
7.1 Übersicht	120
7.2 Bitstopfen.	120
7.3 CAN-Bus Fehlererkennung	121
7.3.1 Bitfehler	122
7.3.2 Bitstopffehler	122
7.3.3 CRC-Fehler.	122
7.3.4 Frame-Fehler	122
7.3.5 ACK-Fehler.	122
7.4 CAN-Bus Fehlerbegrenzung.	123
7.5 Zusammenfassung.	124
Kapitel 8 • CAN-Bus Analyzer	125
8.1 Übersicht	125
8.2 CAN-Bus Analyzer	125
8.2.1 Microchip Inc. CAN bus analyzer.	125
8.2.2 CANdo.	126
8.2.3 PCAN-Explorer	127
8.2.4 CAN-Bus-Tester 2 (CBT2).	128
8.2.5 BitScope Logic	129
8.2.6 LAP-C Logikanalysator.	130
8.3 Projekt 11: CAN-Bus Sniffer	131
8.4 Zusammenfassung.	135

Kapitel 9 • CAN-Busprojekte mit Raspberry Pi	137
9.1 Übersicht	137
9.2 Projekt 11: Einfache CAN-Bus Kommunikation zwischen Raspberry Pi und Arduino Uno	137
9.3 Projekt 12: Raspberry Pi zeigt Nachrichten vom CAN-Bus	144
9.4 Projekt 13: Steuerung der an Arduino Uno angeschlossenen LEDs.	146
9.5 Verwendung von Python für CAN-Bus Programme	149
9.6 Projekt 14: Steuerung der LEDs mit Drucktasten.	150
9.7 Projekt 15: CAN-Bus mit 3 Knoten: Steuerung der LEDs auf den Knoten Raspberry Pi und Arduino Uno	152
Anhang	158
Index	159

Vorwort

Die Struktur des Controller Area Network (Akronym: CAN) wurde ursprünglich für die Anwendung in Personenkraftwagen entwickelt. Heute stehen hochentwickelte CAN-Steuerchips von mehr als 20 Herstellern zur Verfügung, und CAN wird in unterschiedlichen Bereichen wie Medizin, Raumfahrt, Prozesssteuerung, Automatisierung usw. eingesetzt. Seit der Einführung von CAN bei der Organisation ‚CAN in Automation‘ (CiA) bereits im Jahre 1992 haben die Hersteller und Anwender große Anstrengungen unternommen, um die CAN-Standards und CAN-Spezifikationen weiterzuentwickeln.

In diesem Buch wird die Verwendung der Steuerplatinen Arduino Uno und Raspberry Pi 4 mit CAN-Bus Schnittstellenmodulen für die Entwicklung praktischer Projekte auf der Basis von CAN-Bus beschrieben. Beispiele von beliebten Hardware und Software Development Kits werden auf präzise und leicht verständliche Art dargestellt. Die Nutzung dieser Kits vereinfacht den Embedded Design Zyklus erheblich und erleichtert deutlich die Entwicklung, Fehlerbeseitigung und Überprüfung eines Projekts, das auf dem CAN-Bus basiert. Es werden Projekte vorgestellt, welche CAN-Bus Varianten mit 2 und 3 Knoten benutzen.

Geschrieben wurde das vorliegende Buch für Studenten, Ingenieure, Technik-Begeisterte und jeden, der mehr über den CAN-Bus und seine Anwendungen lernen möchte. Die Verfasser gehen davon aus, dass der Leser Grundkenntnisse der Elektronik besitzt. Vorteilhaft sind auch Erfahrungen mit den Programmiersprachen C und Python sowie eine gewisse Vertrautheit mit Arduino Uno und Raspberry Pi, insbesondere dann, wenn der Leser beabsichtigt, Mikrocontroller-Projekte unter Verwendung des CAN-Busses zu entwickeln. Dieses Buch soll eine nützliche Informationsquelle und ein Nachschlagewerk für jeden sein, der Antworten auf eine oder mehrere der folgenden Fragen sucht:

- Welche Bus-Systeme stehen für die Automobilindustrie zur Verfügung?
- Was sind die Grundprinzipien des CAN-Busses?
- Welche Arten von Frames (oder Datenpaketen) stehen in einem CAN-Bussystem zur Verfügung?
- Wie können Fehler in einem CAN-Bussystem erkannt werden, und wie zuverlässig ist ein CAN-Bussystem?
- Welche Arten von CAN-Bus Controllern gibt es?
- Welches sind die Funktionsprinzipien des MCD2515 CAN-Bus Controllers?
- Wie kann ich ein CAN-Bus Project mit Arduino Uno realisieren?
- Wie realisiere ich Arduino CAN-Bus Projekte?
- Wie kann ich die Daten auf dem CAN-Bus analysieren?
- Wie kann ich ein CAN-Bus Projekt mit Raspberry Pi ausführen?
- Wie realisiere ich Raspberry Pi CAN-Bus Projekte mit 2 und 3 Knoten?

Wir hoffen, dass dieses Buch für Sie hilfreich ist, Ihnen Freude macht und Sie in die Lage versetzt, Ihr nächstes CAN-Bus Projekt mit Arduino Uno und/oder Raspberry Pi mit Hilfe von CAN-Bus Schnittstellenmodulen zu realisieren.

*Dogan Ibrahim und Ahmet Ibrahim,
London, 2022*

Kapitel 1 • Kraftfahrzeug-Bussysteme

1.1 Übersicht

Die heutigen Fahrzeuge sind komplexe Maschinen mit mechanischen und elektronischen Bauteilen. Die Anzahl der im Kfz eingebauten elektronischen Komponenten hat in den letzten Jahren stark zugenommen. In der Folge von höheren Anforderungen an Sicherheit, Komfort und Performance kommen in modernen Fahrzeugen immer mehr elektronische Bauteile hinzu. Daraus hat sich ein zunehmender Bedarf daran ergeben, diese elektronischen Baugruppen in solcher Weise zu verbinden, dass sie zuverlässig, sicher und in Echtzeit miteinander kommunizieren können.

In der Vergangenheit waren die elektronischen Komponenten mit hunderten von Leitungen auf komplexe Weise mit den verschiedenen Teilen des Fahrzeugs verbunden. Dementsprechend war es schwer, einen elektronischen Fehler zu finden. Es gab keine Koordination zwischen den verschiedenen elektronischen Baugruppen, da jedes elektronische Element unabhängig von den anderen angesteuert wurde. Wartung und Reparatur der Fahrzeugelektronik waren extrem schwierig, da es in vielen Fällen nicht einfach war, ein fehlerhaftes Bauteil zu lokalisieren und auszutauschen.

Die Fahrgastsicherheit wurde zu einem der wichtigsten Gesichtspunkte in den heutigen Automobilen. Im letzten Jahrzehnt sind Sicherheitseinrichtungen in den Vordergrund getreten und haben sich von physischen zu elektronisch unterstützten Sicherheitssystemen gewandelt. Das reicht von der Bremstechnik über den Kollisionsschutz und Airbags bis zu den sicherheitsrelevanten Fahrerassistenzsystemen der neuesten Generation. Die modernsten Fahrzeuge sind heute intelligente und innovative Maschinen, die mit einer Vielzahl von Sensoren bestückt sind, über die die Umgebung erfasst wird und die dem Fahrer wichtige sicherheitsrelevante Informationen liefern. Diese Sensoren bilden zusammen mit Stellgliedern, Displays und schnellen digitalen Prozessoren wie den spezialisierten high-speed Mikrocontrollern ein intelligentes lokales Netzwerk.

Abbildung 1.1 zeigt ein traditionelles Elektroniksystem eines älteren Fahrzeugs, bei dem Sensoren und Stellglieder auf komplexe Weise miteinander verdrahtet sind. Eines der Hauptprobleme dieser Konzeption liegt in der Wartung. Die Verkabelung war so komplex, dass es nahezu unmöglich war, Störungen aufzufinden und zu korrigieren.

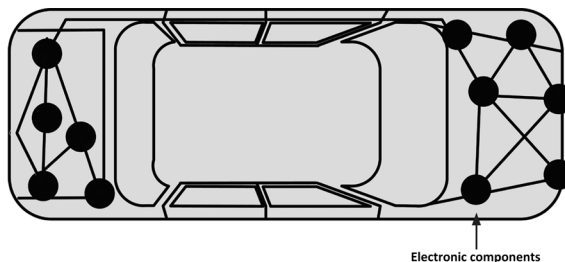


Abbildung 1.1: Traditionelle Fahrzeugelektronik älteren Typs

Mit zunehmender Komplexität der im Fahrzeug eingebauten Elektronik wurde es für die Fahrzeughersteller immer schwieriger, mit den hergebrachten Methoden sichere und zuverlässige Elektroniksysteme zu fertigen. Die heutigen Anforderungen können mit einer simplen elektronischen Steuereinheit nicht mehr erfüllt werden. Die Lösung besteht darin, die verschiedenen elektronischen Module mit einem hocheffizienten Netzwerk zu verzahnen. Deshalb wurde es notwendig, ein Elektroniksystem auf Netzwerkbasis zu konzipieren, in dem elektronische Module leicht in das Netzwerk eingegliedert und dann von einer zentralen intelligenten Steuereinheit (z.B. einer Engine Control Unit) angesteuert werden können. Das Ergebnis ist ein „intelligentes“ Auto, bei dem viele verschiedene Messfühler und Stellglieder genutzt werden, um das Umfeld zu erfassen und multiple Funktionen auszuführen. Beispielhaft dafür ist das automatische Einschalten der Scheinwerfer, wenn es dunkel wird oder das Auto in einen Tunnel einfährt. Als weiteres Beispiel sei die automatische Aktivierung der Scheibenwischer bei einsetzendem Regen genannt.

Einer der Vorteile eines netzwerkbasierten Elektroniksystems in Fahrzeugen liegt darin, dass es relativ einfach ist, ein defektes Modul zu erkennen und aufzufinden. Drüber hinaus ist die Verkabelung wesentlich einfacher und deshalb leichter in Stand zu halten. Zum Beispiel kann man bei der Kommunikation mit der zentralen intelligenten Steuereinheit feststellen, ob die Gesamtelektronik in Ordnung ist und gegebenenfalls ein fehlerhaftes Modul leicht ermitteln. Ein Netzwerksystem erlaubt auch den einzelnen Modulen im BUS untereinander zu kommunizieren und bei Bedarf Informationen auszutauschen. Beispielsweise kann der intelligente Controller die Motortemperatur vom Temperaturfühler-Modul erhalten, die dann auf einem elektronischen Dashboard angezeigt werden kann. Sollte die Temperatur zu hoch sein, werden entsprechende Meldungen an die betreffenden Motorkomponenten geschickt, so dass notwendige Korrekturmaßnahmen eingeleitet werden können. Abbildung 1.2 zeigt ein modernes Fahrzeug, bei dem ein elektronisches Netzwerksystem für die Verbindung und Steuerung der Elektronikbaugruppen eingesetzt wird.

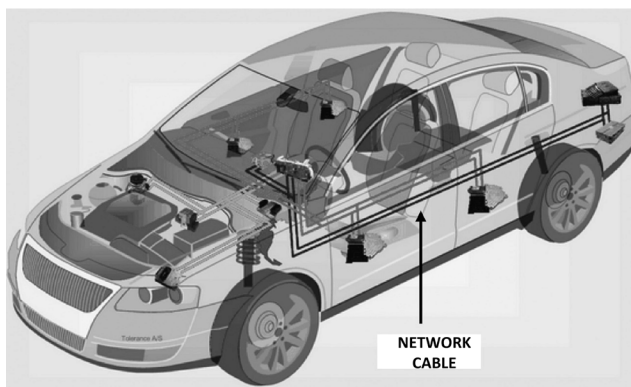


Abbildung 1.2: Modernes Fahrzeug mit elektronischem Netzwerk

Dieses Kapitel gibt eine Übersicht über die wichtigsten in heutigen Fahrzeugen verwendeten Netzwerksysteme. Darüber hinaus enthält es eine Tabelle, in der die Vor- und Nachteile der einzelnen Systeme gegenübergestellt werden.

1.2 Netzwerksysteme in Fahrzeugen

Die in Fahrzeugen verwendeten Netzwerke wurden 1994 von dem Verband der Automobilingenieure SAE (Society of Automotive Engineers) klassifiziert. Entsprechend dieser Klassifikation wurden die Bussysteme anhand ihrer Bandbreite (Datenrate) und Netzwerkfunktionen eingeteilt. Die Klassifikation unterscheidet 4 Kategorien: Klasse A, Klasse B, Klasse C und Klasse D.

Netzwerke der Klasse A sind langsame und kostengünstige Netzwerke mit Datenübertragungsraten von weniger als 10 kbit/s (oder kbps). Sie werden hauptsächlich in Fahrzeugkarosserien eingesetzt.

Netzwerke der Klasse B arbeiten zwischen 10 und 125 kbit/s und werden benutzt für den Informationsaustausch z.B. mit Kombi-Instrumenten für Fahrzeuggeschwindigkeit usw.).

Netzwerke der Klasse C arbeiten zwischen 125 kbit/s und 1 Mbit/s und werden in einem breiten Anwendungsbereich wie der Motorsteuerung eingesetzt.

Netzwerke der Klasse D arbeiten oberhalb von 1 Mbit/s und werden hauptsächlich für Telematikfunktionen verwendet.

Es gibt viele Automobil-Netzwerksysteme (oder Bussysteme, da Fahrzeugnetzwerke in der Form von Busleitungen ausgeführt sind), die entweder von den Fahrzeugherstellern selbst oder in Zusammenarbeit mit Halbleiterfirmen entwickelt wurden. Zu den verbreitetsten Automobil-Netzwerksystemen gehören:

- CAN-Bus
- LIN-Bus
- FlexRay
- MOST
- Byteflight
- DSI-Bus
- Intellibus
- SAE J1850
- BST-Bus
- NML-Bus
- und andere

In diesem Abschnitt werden wir kurz die Details der unten aufgelisteten und am häufigsten benutzten Automobil-Netzwerksysteme betrachten. Wir konzentrieren uns dabei auf den CAN-Bus, der ja Thema dieses Buches ist.

- LIN-Bus
- FlexRay
- MOST
- Byteflight
- Intellibus
- CAN-Bus

1.2.1 LIN-Bus

Der LIN-Bus (Local Interconnect Bus) ist ein preiswerter Bus, der mit 20 kbit/s arbeitet und der entwickelt wurde, um in der Fahrzeugkommunikation einen Standard im kostensparenden unteren Segment zu schaffen. LIN gehört zu den Low-End-Netzwerksystemen im Automobil-Bereich und sie sind kostengünstig. Dieser Bus wird vorwiegend für weniger wichtige Karosserie- oder Komfortfunktionen eingesetzt. Bei LIN handelt es sich um ein 1-Wire-Bussystem mit ‚Single-Master/Multiple-Slave‘ Funktion, wobei die Fahrzeugkarosserie als Rückleitung benutzt wird. Bei einer typischen Anwendung sendet der Master zur Abfrage von Daten eine Nachricht mit einem Header aus, und der Slave mit dem korrekten Nachrichten-Header sendet die angeforderten Daten zusammen mit einer Prüfsumme für die Fehlerprüfung zurück. Zu den typischen Anwendungen des LIN-Bus gehören die Steuerung kleiner Motoren für Scheibenwischer, Schiebedach, Heizregler, Regensensoren, Lenkrad, Sitze, Türen usw., bei denen keine hohe Bandbreite erforderlich ist. Der LIN-Bus wird dort eingesetzt, wo die Verwendung eines CAN-Bus zu teuer wäre. Die ursprünglichen LIN-Spezifikationen wurden von einem Konsortium festgelegt, das aus BMW, Audi, Volvo, VW, Motorola, Volcano und DaimlerChrysler bestand. Der LIN-Bus basiert auf dem Serial Communication Interface (UART) mit 8-Bit Daten.

1.2.2 FlexRay

Mit zunehmenden Forderungen nach mehr Sicherheit, verbesserter Performance und höherem Komfort entsteht auch der Wunsch nach komplexeren und weiterentwickelten elektronischen Fahrzeugnetzwerken. FlexRay wurde 1999 ursprünglich von BMW und DaimlerChrysler als schnelles, effizientes und fehlerfreies Fahrzeug-Bussystem entwickelt. Da es Bandbreiten bis zu 10 Mbit/s erlaubt, ist FlexRay für sehr schnelle Funktionen in Echtzeit gut geeignet. Es können sowohl elektrische als auch optische Übertragungsmedien benutzt werden. FlexRay unterstützt Einkanal- und Zweikanal-Konfigurationen, wobei zwei Kanäle eine höhere Fehlertoleranz und größere Bandbreite bieten. FlexRay wird als nächste Stufe nach LIN und CAN betrachtet und für sicherheitsrelevante Anwendungen und zur schnellen Motorsteuerung in Echtzeit eingesetzt. Zu den Anwendungsgebieten von FlexRay gehören u.a. Motorsteuerung, ABS, Getriebesteuerung, Bremskraftregelung und Fahrwerksteuerung.

1.2.3 MOST

Der MOST-Bus (Media Oriented Systems Transport) wird in Automobilen hauptsächlich für Telemetrie- und Multimedia-Anwendungen wie Audio, Video, Navigation, Kommunikation und dergleichen eingesetzt. In modernen Fahrzeugsystemen befinden sich Multimediageräte wie DVD/CD-Spieler, Fernsehgeräte, Navigationssysteme, Graphikdisplays und mobile Rechner, die sehr hohe Datenübertragungsraten verlangen. Erreicht wird dies mit dem MOST-Netzwerk, das ursprünglich 1998 von BMW und DaimlerChrysler entwickelt wurde.

MOST unterstützt maximal 64 Geräte auf dem Bus mit sehr hoher Bandbreite. Typische Datenübertragungsraten sind 28,8 Mbit/s synchron und 14,4 Mbit/s asynchron. In der Entwicklung befinden sich z.Z. Systeme mit bis zu 50 Mbit/s und sogar 150 Mbit/s. Ein optisches Medium wird für die Datenübertragung benutzt, das frei von elektromagnetischer Strahlung und Interferenzen ist.

1.2.4 Byteflight

Byteflight wurde von BMW entwickelt und bietet eine Bandbreite von 10 Mbit/s. Hauptsächlich eingesetzt wird es in sicherheitsrelevanten Netzwerken in der Automobil- und Luftfahrtindustrie (z.B. bei Fahrzeug-Airbags, Karosserieelektronik usw.). Um hohe Sicherheit zu gewährleisten, muss das Datenprotokoll fehlertolerant und deterministisch sein. Vor Byteflight war die Datenübertragung entweder ereignisgesteuert oder zeitgesteuert. Ereignisgesteuerte Systeme (z.B. CAN) übertragen Daten nur, wenn die Daten bereitstehen oder wenn eine Datenabfrage erfolgt. Zeitgesteuerte Datenprotokolle räumen jedem Knoten ein Zeitfenster entsprechend einer zuvor festgelegten Sequenz ein. Die Zahl der zu übertragenden Nachrichten kann während des Betriebs im Allgemeinen nicht verändert werden, da die Anzahl der zugeordneten Zeitfenster feststeht. Das Byteflight Protokoll kombiniert sowohl ereignisgesteuerte als auch zeitgesteuerte Protokolle, und es garantiert deterministische Latenzzeiten für eine spezifische Anzahl von Nachrichten hoher Priorität, und die flexible Nutzung von Bandbreiten für Nachrichten niedriger Priorität.

Erstmals in der Serienfertigung eingesetzt wurde Byteflight von BMW in der 7er Reihe für eine vernetzte passive Sicherheitsanwendung. Byteflight unterstützt verschiedene Netzwerkprotokolle in einem gemischten Bus-Umfeld.

1.2.5 Intellibus

Intellibus ist ein Multi-Drop Kommunikationsbus hoher Schnelligkeit, der eine Bandbreite bis zu 15 Mbit/s zulässt. Er wurde ursprünglich von Boeing konzipiert, um die komplexe Verkabelung der in Flugzeugen verteilten Systeme zu vereinfachen. Es handelt sich dabei um einen preiswerten Bus, der den Anschluss einer großen Zahl von Sensoren ermöglicht und überwiegend für die Motorsteuerung, Transmission und andere Komponenten, die eine hohe Geschwindigkeit verlangen, eingesetzt wird. Ein Intellibus-Netzwerk besteht typischerweise aus einem Network Interface Controller (NIC) und 1 bis 255 Intellibus Interface-Modulen (IBIM), die mit Sensoren verbunden werden können. Der NIC kann an einen PC oder ein anderes elektronisches Gerät angeschlossen werden. Normalerweise wird der NIC mit Software heruntergeladen, um Daten von verschiedenen IBIMs in planmäßiger Weise in spezifischen Intervallen zu erfassen. In einem komplexen Netzwerk können zwei oder mehrere NIC-Karten verwendet werden, um die Knotenkapazität zu erhöhen. Für die Programmierung des NIC und der IBIMs steht spezielle Software zur Verfügung. Die maximale Buslänge beträgt 30 m (bei 12,5 MBit/s und 64 Knoten). Der Intellibus wird eingesetzt in der Fahrzeugelektronik, Prozesssteuerung, Automatisierung, Luftfahrt, Medizintechnik und mehreren weiteren Gebieten.

1.2.6 CAN-Bus

Der CAN-Bus (Controller Area Network) ist das Hauptthema in diesem Buch. Deshalb untersuchen wir in diesem Abschnitt die grundsätzlichen Eigenschaften des CAN-Busses. Die weiteren Kapitel des Buchs enthalten detaillierte Beschreibungen von CAN-Bus Anwendungen und von Projekten im Zusammenhang mit CAN-Bus.

CAN ist ein serieller zweiadriger Multimaster-Bus, der in den 1980ern von der Robert Bosch GmbH entwickelt wurde. Er ist heute einer der am weitesten verbreiteten Feldbusse in der

Fahrzeugindustrie. Die physikalische Schicht besteht aus einem verdrehten Leitungspaar. CAN erlaubt die zuverlässige, robuste und schnelle Kommunikation mit bis zu 1 Mbit/s (mit 40 m Buslänge). CAN 2.0A ist die originale CAN-Version mit den folgenden Feldern:

- Start-of Frame Bit
- 18 Bit Header (mit 11 Bit Nachrichten-Identifizier)
- 0-8 Byte Daten
- 15 Bit Cycle Redundancy Check (CRC)
- 3 Bit Acknowledgement Slot
- 7 Bit End of Frame

Der CAN beruht auf dem CSMA/CR-Mechanismus (Carrier Sense Multiple Access/Collision Resolution), um während der Datenübertragung auf dem Bus Framekollisionen zu vermeiden. Jeder CAN-Knoten überwacht den Bus, und sobald der Knoten feststellt, dass der Bus frei ist, kann die Datenübertragung begonnen werden. Falls andere Knoten auf dem Bus zur gleichen Zeit versuchen, Daten zu übermitteln, findet eine Arbitrierung statt, und der Knoten mit der höchsten Priorität (d.h. mit dem niedrigsten Nachrichten-Identifizier) erhält das OK und sendet seine eigenen Daten. Der CAN-Bus verfügt über einen einfachen Mechanismus zur Fehlererkennung und Fehlerbeseitigung. Empfangende Knoten überprüfen die Integrität der Nachrichten, indem sie die CRC-Felder durchsehen. Wird ein Fehler festgestellt, werden die anderen Knoten auf dem Bus durch eine Error Flag (Fehlererkennung) benachrichtigt. Abbildung 1.3 zeigt eine typische CAN-Bus Anwendung mit den zwei Knoten A und B. CAN ist ein Netzwerk der Klasse A/B.

Zusammenfassung der CAN-Bus Eigenschaften:

- Multimaster Bus, bei dem jeder Knoten ein Master sein kann
- Multicast-Empfang
- Geschwindigkeiten bis zu 1 Mbit/s
- Hohe Zuverlässigkeit

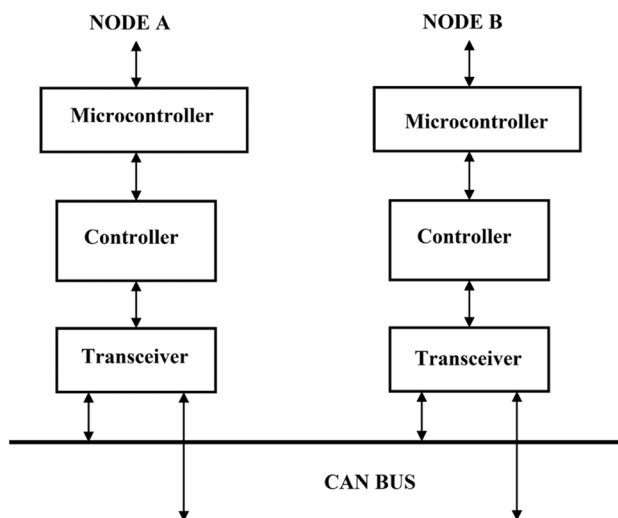


Abbildung 1.3: CAN-Bus mit zwei Knoten

1.2.7 Weitere Busstandards

Weitere Bus-standards, die in diesem Kapitel erwähnt werden sollen, sind der 1984 entwickelte SAE J1850 (oder einfach J1850), und der MI-Bus (Motorola Interconnect). J1850 fand verbreitete Anwendung in den Automobilen von GM, Chrysler und Ford, und diente der Diagnostik und dem Datenaustausch. Es gibt zwei Versionen dieses Standards: PWM (Pulse Width Modulation) mit 41,6 kbit/s und einer zweiadrigen differentiellen physikalischen Schicht und VPW (Variable Pulse Width) mit 10,4 kbit/s und einer einadrigen physikalischen Schicht. Die beiden Standards sind miteinander nicht kompatibel. Das J1850 Protokoll besitzt einen Frame-Aufbau und benutzt CSMA/CR-Arbitrierung. Ein Frame besteht aus einem Start of Frame, Header-Byte, Datenbytes, einem Byte CRC und einem End of Data Symbol (ein 200 μ s Impuls). Die meisten OBD-Tools (On Board Diagnostic) unterstützen das J1850 Protokoll für diagnostische Zwecke. Der J1850 Standard ist alt und läuft aus. Bei dem J1850 Standard handelt es sich um ein Netzwerk der Klasse B.

Der MI-Bus ist ein einadriger Bus mit einem Master und mehreren Slaves. Es handelt sich um einen preiswerten Bus mit niedriger Datenrate, der hauptsächlich zum Antrieb von Sitzen, Spiegeln usw. dient. Master ist der Controller, der Adressen und Daten an alle Slaves auf dem Bus sendet. Slaves mit den passenden Adressen reagieren auf die Anforderung.

1.3 Vergleich der Bussysteme in Fahrzeugen

Heute ist der CAN-Bus, der mit Geschwindigkeiten bis zu 1 Mbit/s arbeiten kann, das am häufigsten verwendete Bussystem.

Tabelle 1 teigt eine Zusammenstellung der üblicherweise verwendeten Fahrzeugnetzwerke (oder Bussysteme). Verglichen werden darin die verschiedenen Systeme. Für jedes System werden Klasse, allgemeine Informationen, Bandbreite und typischen Anwendungen angegeben.

Allgemein wird der LIN-Bus in der Low-Speed Automatisierung angewendet wie für Scheibenwischer, Regensensoren und dergleichen. FlexRay hingegen wird für sehr leistungsfähige und sicherheitsrelevante Funktionen benutzt. Es handelt sich dabei um eine relativ neue Busstruktur, und man geht davon aus, dass sie der Automobilstandard der Zukunft sein wird. MOST findet in Fahrzeugen häufig Anwendung bei Multimedia- und Navigationstechniken. Dieser Bus benutzt optische Leitungen und ist unempfindlich für elektrische Störsignale. Byteflight ist ebenfalls ein neues Fahrzeug-Netzwerksystem und gut geeignet für leistungsfähige, sehr schnelle und sicherheitsrelevante Anwendungen wie die Steuerung von Airbags. Byteflight benutzt Glasfaseroptik und ist damit unempfindlicher gegenüber elektromagnetischer Strahlung und elektrischem Rauschen. Intellibus schließlich bietet eine hohe Bandbreite und wird vorwiegend in der Luftfahrt angewendet.

	Klasse	Allgemeines	Bandbreite	Anwendung
CAN	A, B	verdrilltes Leitungspaar geringe Kosten breite Anwendung mehrfache Master	1 Mbit/s	- Karosserie - Motor
LIN	A	Geringe Kosten niedrige Geschwindigkeit Master/Slave einadrig	10 kbit/s	- Karosserie - Komfort
FlexRay	D	sehr hohe Geschwindigkeit zukünftiger Standard Master/Slave (für Synchronisierung)	10 Mbit/s	- Motor - Sicherheit
MOST	D	Dateneffizienz Standard für Multimedia Master/Slave (für Synchronisierung) Glasfaseroptik	25 Mbit/s	Multimedia
Byteflight	D	hohe Geschwindigkeit Master/Slave Glasfaseroptik	10 Mbit/s	Sicherheitsrelevanz
Intellibus	D	hohe Geschwindigkeit	15 Mbit/s	- Karosserie - Motor - Sicherheit

Tabelle 1.1: Fahrzeug-Bussysteme

Die Abbildungen 1.4 und 1.5 zeigen Beispiele von Personenkraftwagen, in denen mehrere Bussysteme verwendet werden (Quelle: *Introduction to CAN – Application Note REJ05B0804-0100/Rev. 1.00*). In Abbildung 1.4 ist der CAN-Bus links in der Abbildung zusehen. Er ist verbunden mit Instrumenten, Klimaregelung, Beleuchtung, Lenkrad, Antriebssystem und Verriegelung. Der MOST-Bus ist im linken unteren Teil der Abbildung zu sehen. Er verbindet Multimediageräte wie Lautsprecher, Digitalradio, Fahrzeugcomputer und Navigationsgerät. In dieser speziellen Konfiguration ist FlexRay unten rechts in der Abbildung zu sehen, verbunden mit Motor, Lenksystem und Bremsen. Ein Diagnosegerät, das rechts unten im Bild an den CAN-Bus angeschlossen ist, dient der Überprüfung des Fahrzeugzustands.

In Abbildung 1.5 werden Scheinwerfer, Klimaanlage, Scheibenwischer, Türen und Fenster über den LIN-Bus gesteuert, während ein Flexbus die Bremsen versorgt. Die meisten anderen Einrichtungen des Fahrzeugs werden über den CAN-Bus gesteuert.