



Marko Domijanec, bacc. ing. comp.

## **Ethernet im Kraftfahrzeug**

### **MASTERARBEIT**

zur Erlangung des akademischen Grades

Diplom-Ingenieur

Masterstudium Telematik

eingereicht an der

**Technischen Universität Graz**

Betreuer

Ass.Prof. Dipl.-Ing. Dr.techn. Bernd Eichberger

Institut für Elektronische Sensorsysteme

Zweitbetreuer: Dipl.-Ing. Dr.techn. Martin Dohr



## **EIDESSTATTLICHE ERKLÄRUNG**

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Textdokument ist mit der vorliegenden Masterarbeit identisch.

---

Datum

---

Unterschrift





# Danksagung

Danken möchte ich in erster Linie meinen Betreuern Herrn Prof. Bernd Eichberger und Dr. Martin Dohr, die mich tatkräftig und mit Geduld kontinuierlich bei der Masterarbeit unterstützt haben.

Mein Dank gilt der MAGNA STEYR Engineering, die diese Masterarbeit ermöglicht hat. Weiters allen Mitarbeitern, die mich in irgendeiner Weise unterstützt haben und besonders Herrn Wolfgang Reinprecht für die produktiven Gespräche.

Für die Hilfe bei der Durchführung der praktischen Arbeit möchte ich mich bei Herrn Eduard Dorner und seinem Team vom Institut für Elektronik der Technischen Universität Graz bedanken.

Ganz herzlich möchte ich mich bei meinen Eltern, meiner Schwester und Großmutter bedanken, die mich bedingungslos über das ganze Studium in jeder Form unterstützt haben.

Zuletzt gilt mein Dank allen Freunden und Kollegen, die mir während des Studiums und der Masterarbeit auf verschiedenste Weise behilflich waren.



## Kurzfassung

Durch die immer wachsende Anzahl von Steuergeräten und die zunehmende Vernetzung des Fahrzeugs intern und mit seiner Umgebung, steigen die Anforderungen an die Vernetzungs-Technologien im Fahrzeug. Vorhandene Bussysteme sind diesen Anforderungen in Bezug auf Bandbreite und Wirtschaftlichkeit nicht mehr gewachsen.

Ethernet hat im industriellen Netzwerksektor seine Zuverlässigkeit und Qualität unter extremen Bedingungen bewiesen. Jetzt bietet es sich als kostengünstige und verbreitete Lösung für den Einsatz im Kraftfahrzeug an. Doch ist Ethernet dafür wirklich geeignet?

Ziel dieser Masterarbeit ist eine umfassende Untersuchung des Themas „automotives Ethernet“, sowie die Bewertung der Einsatztauglichkeit von Ethernet im Kraftfahrzeug.

Im theoretischen Teil werden die Grundlagen der Netzwerk-Kommunikation und der Ethernet-Technologie, wie auch die am häufigsten eingesetzten Bussysteme im Kraftfahrzeug beschrieben. Weiters wird die Motivation hinter der Einführung von Ethernet in das Auto erläutert und eine Recherche zum Stand der Technik durchgeführt.

Der praktische Teil wird in zwei Arbeitspakete unterteilt. Im ersten Arbeitspaket wird mithilfe zweier Evaluierungsplattformen ein Demonstrator mit Standard-Ethernet-Bausteinen, hard- und softwaremäßig in Betrieb genommen. Dabei wird eine Schnittstelle zwischen CAN und Ethernet realisiert. Diese empfängt Nachrichten vom CAN-Modul, setzt sie auf ein Ethernet-taugliches Protokoll (UDP/TCP) um und sendet diese über den Ethernet-Transceiver zur Empfängerplatine. Die Empfängerplatine arbeitet den Signalfluss in der umgekehrten Richtung ab.

Im zweiten Arbeitspaket werden die Standard-Ethernet-Transceiver durch seriennahe Automotive-Ethernet-Transceiver (BroadR-Reach®) ersetzt und auf die vorhandene Hardware- und Softwarestruktur adaptiert.

Anschließend werden standardmäßige Untersuchungen (EMV, Signalqualität) für Technologien im Fahrzeug durchgeführt, welche die automobilen Tauglichkeit der Ethernet-Technologie bestätigen.

## Abstract

Due to the steadily growing number of ECUs and the increasing connectivity of the vehicle internally and with its environment, the demands on the networking technologies are increasing. Existing bus systems are no longer up to these requirements in terms of bandwidth and cost-effectiveness.

In the industrial network sector, Ethernet has proven its reliability and quality under extreme conditions. Now it offers itself as a cost-effective and popular solution for use in motor vehicles. But is Ethernet really suitable for it?

The aim of this master thesis is a comprehensive investigation of the topic „Automotive Ethernet“, as well as the evaluation of the suitability of Ethernet in motor vehicles.

The theoretical part describes the fundamentals of network communication and Ethernet technology, as well as the most commonly used bus systems in motor vehicles. Furthermore, the motivation behind the introduction of automotive Ethernet is explained and a research of the state of the art is performed.

The practical part is divided into two work packages. In the first work package, with the help of two evaluation platforms, a demonstrator with standard Ethernet components is set up. For that, an interface between CAN and Ethernet is realized. The interface receives messages from the CAN module, converts them to an Ethernet suitable protocol (UDP/TCP) and sends them via the Ethernet transceiver to the receiver board. The receiver board processes the signal flow in the opposite direction again.

In the second work package, standard Ethernet transceivers are replaced by close to production automotive Ethernet transceivers (BroadR-Reach®) and adapted to the existing hardware and software structure.

Subsequently, standard investigations (EMC, signal quality) for automotive technologies are carried out, demonstrating the automotive suitability of the Ethernet technology.

# Inhaltsverzeichnis

Abstrakt	vii
Abbildungsverzeichnis	xv
Tabellenverzeichnis	xvii
Abkürzungsverzeichnis	xxi
1. Einleitung	1
1.1. Aufbau der Arbeit . . . . .	2
2. Grundlagen	3
2.1. OSI-Referenzmodell . . . . .	3
2.1.1. Transport Layer . . . . .	8
2.1.1.1. UDP . . . . .	8
2.1.1.2. TCP . . . . .	9
2.2. Ethernet . . . . .	12
2.2.1. Standards . . . . .	14
2.2.1.1. 10 Mbit/s Ethernet . . . . .	14
2.2.1.1.1. CSMA/CD . . . . .	16
2.2.1.1.2. Ethernet Rahmen . . . . .	18
2.2.1.2. Fast Ethernet . . . . .	19
2.2.2. Topologien . . . . .	20
2.2.2.1. Bus-Topologie . . . . .	21
2.2.2.2. Stern-Topologie . . . . .	21
3. Ethernet im industriellen Bereich	23
3.1. Industrial Ethernet . . . . .	23
3.1.1. Anforderungen . . . . .	23
3.1.2. Lösungskonzepte . . . . .	24
3.1.2.1. PROFINET . . . . .	25
3.1.2.2. EtherCAT . . . . .	27
3.1.2.3. Modbus-TCP . . . . .	28
3.1.2.4. Ethernet POWERLINK . . . . .	29
3.1.2.5. EtherNet/IP . . . . .	31
3.1.3. Datenbussysteme im Vergleich . . . . .	32
3.1.4. Safety Aspekte . . . . .	34
3.1.5. Embedded TCP/IP-Stacks . . . . .	34
3.1.5.1. Open-Source-Implementierungen . . . . .	36
3.1.5.2. Kommerzielle Implementierungen . . . . .	37

## Inhaltsverzeichnis

3.2. Industrial Wireless . . . . .	38
3.2.1. Wireless-Technologien . . . . .	39
4. Bussysteme im Fahrzeug . . . . .	41
4.1. CAN-Bus . . . . .	41
4.1.1. Physical Layer . . . . .	42
4.1.2. Fehlererkennung und -behandlung . . . . .	43
4.1.3. Nachrichtenformat . . . . .	44
4.1.4. CAN FD . . . . .	45
4.2. LIN-Bus . . . . .	47
4.2.1. Eigenschaften und Buszugriffsverfahren . . . . .	47
4.2.2. Nachrichtenformat . . . . .	48
4.3. FlexRay . . . . .	49
4.3.1. Bus-Topologien . . . . .	49
4.3.2. Physical Layer . . . . .	50
4.3.3. Nachrichtenformat und Buszugriffsverfahren . . . . .	51
4.4. MOST . . . . .	53
4.4.1. Physical Layer und Bus-Topologien . . . . .	53
4.4.2. Buszugriffsverfahren und Nachrichtenformat . . . . .	54
5. Ethernet im Kraftfahrzeug . . . . .	57
5.1. Motivation . . . . .	57
5.1.1. Wieso wird Ethernet erst jetzt in Fahrzeugen eingesetzt? . . . . .	59
5.2. Zukünftige Fahrzeugtechnologien . . . . .	60
5.3. Jetzige Automotive Ethernet Architektur . . . . .	61
5.4. OPEN Alliance SIG . . . . .	62
5.4.1. Broadr-Reach® . . . . .	63
5.4.1.1. Physical Layer . . . . .	64
5.5. Echtzeitanwendungen mit Automotive-Ethernet . . . . .	66
5.5.1. Echtzeitfähigkeit mit AVB . . . . .	67
5.5.2. Time-Sensitive Networking . . . . .	69
5.5.3. Interspersed Express Traffic . . . . .	72
5.6. Enabling Technologies . . . . .	73
5.6.1. DoIP . . . . .	73
5.6.2. PoE . . . . .	74
5.6.3. Energy-Efficient Ethernet . . . . .	75
5.7. Teststandards für Automotive-Ethernet . . . . .	76
5.8. Ausblick . . . . .	78
5.8.1. Netzwerkarchitektur mit Automotive-Ethernet . . . . .	79
5.8.2. Marktpotenzial . . . . .	81
6. Praktische Arbeit . . . . .	85
6.1. Ziele . . . . .	85

6.2. Demonstrator - Standard Ethernet . . . . .	86
6.2.1. Verwendete Werkzeuge . . . . .	87
6.2.2. Softwareentwicklung . . . . .	88
6.2.2.1. Design und Initialisierung vom lwIP . . . . .	89
6.2.3. Aufbau und Inbetriebnahme . . . . .	91
6.3. Demonstrator - Automotive Ethernet . . . . .	94
6.3.1. Entwurf und Erstellung der Platine . . . . .	95
6.3.2. Software-Erweiterung . . . . .	99
6.3.3. Aufbau und Inbetriebnahme . . . . .	100
6.3.4. Signalqualität Messungen . . . . .	103
6.3.5. EMV Untersuchung . . . . .	104
7. Zusammenfassung	107
Literatur	111
A. Anhang	117





# Abbildungsverzeichnis

2.1. Vergleich des TCP/IP-, Hybriden- und OSI-Referenzmodells . . . . .	4
2.2. Datenkapselung am Beispiel des hybriden-Referenzmodells bei vertikaler Kommunikation . . . . .	6
2.3. UDP-Segments inkl. Pseudo-Header . . . . .	9
2.4. TCP-Segments inkl. Pseudo-Header . . . . .	10
2.5. TCP-Dreiwege-Handshake . . . . .	11
2.6. IEEE 802 LAN- und MAN-Standards . . . . .	12
2.7. Einordnung der Ethernet-Netzarchitektur in das OSI-Referenzmodell . . . . .	13
2.8. Voll- und Halbduplex Modus . . . . .	16
2.9. Flussdiagramm des Sendevorgangs aus der Sicht eines Netzwerkgerätes . . . . .	17
2.10. IEEE 802.3 Ethernet-Rahmen mit VLAN-Tag . . . . .	18
2.11. Ethernet-Bus-Topologie mit RG58 Koaxialkabel als Übertragungsmedium. . . . .	21
2.12. Ethernet-Stern-Topologie . . . . .	22
3.1. Die Echtzeit-Ethernet-Konzepte werden in drei Kategorien unterteilt . . . . .	25
3.2. PROFINET nutzt eine dreistufige Echtzeitskalierung . . . . .	27
3.3. EtherCAT-Slaves haben einen sehr schlanken Protokoll-Stack, der beliebige andere Protokolle tunneln kann . . . . .	28
3.4. Modbus-TCP Architektur . . . . .	29
3.5. Beim POWERLINK-Protokollstapel ist CANopen das Bindeglied zu anderen Systemen . . . . .	30
3.6. Schichtenmodell des EtherNet/IP-Protokollstapel . . . . .	31
3.7. Unterschiede zwischen industriellen Bussystemen . . . . .	33
3.8. Industrielle Ethernet-Protokolle zur Übertragung von sicherheitsrelevanten Daten . . . . .	34
3.9. Struktur eines Modulare TCP/IP-Stacks für Embedded-Systeme . . . . .	35
3.10. Anwendungsbereiche von Funksystemen in der industriellen Umgebung . . . . .	38
3.11. Drahtlose Erweiterungsmöglichkeiten eines Industriellen Ethernet-Netzwerkes . . . . .	39
4.1. Die wichtigsten CAN-Standards . . . . .	41
4.2. High-Speed-CAN-Netzwerk mit Abschlusswiderständen . . . . .	42
4.3. Signalpegel bei High- und Low-Speed-CAN nach ISO 11898-2 und -3 . . . . .	43
4.4. Botschaftsaufbau des CAN-Standards . . . . .	44
4.5. Botschaftsaufbau des CAN FD . . . . .	45
4.6. Systemtopologie eines LIN-Bussystems . . . . .	47
4.7. LIN-Kommunikationsablauf . . . . .	48
4.8. LIN-Botschaftsformat . . . . .	48
4.9. Beispiele für FlexRay-Topologien . . . . .	50
4.10. Signalpegel auf den FlexRay-Leitungen BP und BM . . . . .	50
4.11. FlexRay Nachrichtenformat . . . . .	51

## Abbildungsverzeichnis

4.12. Kommunikationsablauf bei FlexRay . . . . .	52
4.13. MOST Ring-Struktur-Topologie . . . . .	54
4.14. MOST Nachrichtenformat . . . . .	55
4.15. Übersicht der im Kfz eingesetzten Bussysteme . . . . .	56
5.1. Steigende Anforderungen an die Bandbreite treiben die stetige Optimierung der vorhandenen Technologien und die Einführung von neuen . . . . .	57
5.2. Ein typischer Kabelbaum im Auto . . . . .	58
5.3. Dank Ethernet wird die Umstellung vom heterogenen ins homogene Netz- werk gelingen . . . . .	59
5.4. Automotive Ethernet Architektur in 2015 . . . . .	61
5.5. One-Pair Ethernet BroadR-Reach® Technologie . . . . .	62
5.6. Verkabelungsvergleich zwischen LVDS und BroadR-Reach® . . . . .	63
5.7. 100 Mbit/s BroadR-Reach® Technologie . . . . .	64
5.8. Unterschiede zwischen den IEEE 100Base-TX, 1000Base-T und BroadR- Reach® physikalischen Schichten . . . . .	65
5.9. Echtzeit-Übertragung von Audio- und Videodaten mit AVB . . . . .	67
5.10. Einrichtung einer Echtzeitverbindung mit dem STR . . . . .	68
5.11. Anwendung der IEEE 802.1 AVB Norm in einem Kamerasystem . . . . .	69
5.12. IEEE-802.1Q-Warteschlangen-Modell mit Kommunikations-Schedules („Gate Event List“) . . . . .	70
5.13. IEEE 802.1Qbv Kommunikations-Szenario . . . . .	71
5.14. Ethernet-Schnittstelle zum Fahrzeug . . . . .	73
5.15. DoIP-Transportprotokoll im OSI-Schichtenmodell . . . . .	73
5.16. Schaltplan der <i>Power over BroadR-Reach® Ethernet</i> Technologie . . . . .	74
5.17. Anwendungsfall: <i>Power over BroadR-Reach® Ethernet</i> . . . . .	75
5.18. EEE: Selektiver <i>wake up</i> Mechanismus durch EDM . . . . .	76
5.19. Entwicklung vom Automotive-Ethernet bis 2020 . . . . .	78
5.20. Automotive Netzwerkarchitektur von 2015-2017 . . . . .	79
5.21. Voraussichtliche Automotive Netzwerkarchitektur in 2020 . . . . .	80
5.22. Verschiedene Backbone-Architekturen mit Automotive Ethernet . . . . .	80
5.23. Anzahl der weltweit verkauften Neuwagen über die vergangenen Jahre . . . . .	81
5.24. Vergleich der relativen Kommunikationskosten pro Knoten im Fahrzeug . . . . .	82
5.25. Voraussichtliche Anzahl von Ethernet-Ports im Fahrzeug bis 2019 . . . . .	82
6.1. Datenfluss des Standard-Ethernet-Demonstrators . . . . .	86
6.2. Evaluierungsboard TMS570LS31x . . . . .	87
6.3. Design der lwIP Implementation . . . . .	89
6.4. Projekt Struktur des Standard-Ethernet-Demonstrators . . . . .	90
6.5. Aufbau des Standard-Ethernet-Demonstrators . . . . .	92
6.6. Verbindungs-Aufbau und Daten-Übertragung . . . . .	93
6.7. Datenfluss des BroadR-Reach-Ethernet Demonstrators . . . . .	94
6.8. Pinout Diagramm des BCM89810-Transceiver . . . . .	95

6.9. Ober- und Unter-seite der fertigen Platine . . . . .	96
6.10. Schaltplan der entworfenen Platine . . . . .	97
6.11. Layout der entworfenen Platine . . . . .	98
6.12. Die verwendeten MII-Signale im MII-Lite-Modus zwischen der Evaluierungsboard- und PHY-MII-Schnittstelle . . . . .	99
6.13. Ungeschirmtes-verdrilltes-Zweidraht-Kabel . . . . .	100
6.14. Aufbau der 100 Mbit/s BroadR-Reach-Verbindung im Terminal-Fenster . . .	101
6.15. Aufbau und Inbetriebnahme des BroadR-Reach-Demonstrators im 100 Mbit/s Betrieb . . . . .	102
6.16. Signalqualität bei verschiedenen Kabellängen . . . . .	103
6.17. Aufbau einer leitungsgeführten HF-Störfestigkeitsprüfung . . . . .	105
6.18. Veränderung der Signalqualität (SQI) durch Erhöhung der Störgröße (MHz)	106
A.1. Materialliste der Platine . . . . .	118



# Tabellenverzeichnis

2.1. Übersicht der Protokolle zugeordnet in dazugehörige Schichten des OSI-Referenzmodells[[20]] . . . . .	7
2.2. 10 Mbit/s Ethernet-Schnittstellen[[23]] . . . . .	15
2.3. Fast Ethernet-Schnittstellen[[23]] . . . . .	19
4.1. Fehlermechanismen beim CAN-Bus[[27]] . . . . .	44
6.1. Konnektoren am TMS570-Evaluierungsboard . . . . .	86
6.2. Beschreibung der einzelnen SQI-Werte . . . . .	103
6.3. SQI Werte bei verschiedenen Kabellängen . . . . .	104



# Abkürzungsverzeichnis

IEEE	... Institute of Electrical and Electronics Engineers
ISO	... International Organization for Standardization
OSI	... Open Systems Interconnection
LAN	... Local Area Network
MAN	... Metropolitan Area Network
WLAN	... Wireless Local Area Network
IP	... Internet Protocol
PHY	... Physical Layer
CSMA/CD	... Carrier Sense Multiple Access/Collision Detection
CSMA/CR	... Carrier Sense Multiple Access/Collision Resolution
MAC	... Media Access Control
MII	... Medium Independent Interface
MDI	... Medium Dependent Interface
PLS	... Physical Layer Signaling
PMA	... Physical Medium Attachment
PMD	... Physical Medium Dependent
PCS	... Physical Coding Sublayer
LLC	... Logical Link Control
SFD	... Start Frame Delimiter
RAM	... Random Access Memory
ROM	... Read Only Memory
ARP	... Address Resolution Protocol
PPP	... Point-to-Point Protocol
IGMP	... Internet Group Management Protocol
ICMP	... Internet Control Message Protocol
RFC	... Request for Comments
RTOS	... Real-Time Operating System
CBA	... Component Based Automation
FSoE	... FailSafe over EtherCAT
EPSPG	... Ethernet Powerlink Standardization Group
ODVA	... Open DeviceNet Vendor Association
CIP	... Common Industrial Protocol
ETG	... EtherCAT Technology Group
CoE	... CAN over EtherCAT
EoE	... Ethernet over EtherCAT
FoE	... File Access over EtherCAT
PAD	... Padding-Bits
OS	... Operating System
TI	... Texas Instruments
PC	... Personal Computer

## Abkürzungsverzeichnis

PoE	... Power over Ethernet
C2C	... Car2Car
TCP	... Transmission Control Protocol
ADAS	... Advanced Driver Assistance Systems
UDP	... User Datagram Protocol
UTP	... Unshielded Twisted Pair
PCB	... Printed Circuit Board
BOM	... Bill of Material
CRC	... Cyclic Redundancy Check
KFZ	... Kraftfahrzeug
PKW	... Personenkraftwagen
NKW	... Nutzkraftwagen
LKW	... Lastkraftwagen
HART	... Highway Addressable Remote Transducer
HART	... Multiple Input Multiple Output
SMS	... Short Message Service
AVB	... Audio Video Bridging
C2C	... Car2Car
CAN	... Controller Area Network
PAM3	... Pulse Amplitude Modulation 3
IET	... Interspersed Express Traffic
IPG	... InterPacket Gap
EDM	... Energy Detect Module
OBD	... On Board Diagnose
UDS	... Unified Diagnostic Services
DLC	... Data Length Code
CAN FD	... CAN with Flexible Data-Rate
EDL	... Extended Data Length
BRS	... Bit Rate Switch
TDMA	... Time Division Multiple Access
NIT	... Network Idle Time
TDM	... Time-Division-Multiplex
TTCAN	... Time-Triggered CAN
TSN	... Time-Sensitive Networking
SRP	... Stream Reservation Protocol
EEE	... Energy-Efficient Ethernet
PoDL	... 1-Pair Power over Data Lines
FTDMA	... Flexible Time Division Multiple Acces
FTDMA	... Advanced Driver Assistance Systems
DoIP	... Diagnostics over Internet Protocol
TCP/IP	... Transmission Control Protocol/Internet Protocol
RTP	... Real Time Transport Protocol



RTPGE	... Reduced Twisted Pair Gigabit Ethernet
LIN	... Local Interconnect Network
MOST	... Media Oriented System Transport
LVDS	... Low Voltage Differential Signaling
OEM	... Original Equipment Manufacturer
SIG	... Special Interest Group
ID	... Identifier
QoS	... Quality-of-Service
POF	... Plastic Optical Fiber
FOT	... Fiber Optical Transceiver
lwIP	... light-weight Internet-Protocol
EMAC	... Ethernet Media Access Controller
PLL	... Phase-Locked-Loop
CCS	... Code Composer Studio
MDIO	... Management Data Input/Output
MCU	... Mikrocontroller
LDS	... Long Distance Signaling
EMV	... Elektromagnetische Verträglichkeit
RFI	... Radio Frequency Interference
SF	... Störfestigkeitsprüfung
SQI	... Signal Quality Indicator
MA	... Masterarbeit
z.B.	... zum Beispiel
Tab.	... Tabelle
d.h.	... das heißt
usw.	... und so weiter
bzw.	... beziehungsweise
HALCoGen	... Hercules Peripheral Drivers



# 1. Einleitung

Bedingt durch die stetig wachsende Zahl an Steuergeräten (derzeit bis zu hundert) und der zunehmenden Vernetzung fahrzeugintern sowie mit der Fahrzeugumgebung (Car2X), steigen die Anforderungen an die Vernetzungs-Technologien im Fahrzeug. Vorhandene Bussysteme (CAN, Flexray, MOST und LVDS) sind zu unflexibel und können mit der wachsenden Netzwerk-Komplexität und den immer höheren Bandbreiten-Anforderungen nicht mithalten.

Dies war der Anlass um den von der Automobilindustrie bis jetzt nur außerhalb des Fahrzeugs eingesetzten (Schnittstelle für Diagnostik und *firmware updates*) Ethernet Standard ins Automobil einzuführen.

Neben den geringen Kosten bietet Ethernet ebenfalls eine hohe Bandbreite. Zurzeit sind 100 Mbit/s im Fahrzeug Standard, währenddessen bewegt man sich im Consumer-Bereich (Standard Ethernet) in Richtung 100 Gbit/s, weshalb es viel Raum für Weiterentwicklung gibt.

Ethernet erleichtert den Einzug der Consumer Electronic (einfache Plug & Play Anbindung von Endgeräten und Applikationen) ins Fahrzeug. Mobile Endgeräte von Fahrzeuginsassen sollen Zugang zum Fahrzeugnetz bekommen und somit Multimedia (Video, Musik)- und Kommunikationsinhalte (Telefon, Mail, SMS) mit dem internen Fahrzeug-Multimediasystem nutzen können.

Da es sich bei Ethernet um eine bewährte und weit verbreitete Technologie handelt, steht eine große Anzahl an verwendbaren Protokollen und Applikationen aus der PC- und Industrie-Ethernet-Welt zur Verfügung. Weiters sind viele Entwicklungs- und Diagnosewerkzeuge bereits vorhanden. Durch die Verwendung von Switches, bietet Ethernet eine einfache Skalierbarkeit des Netzwerks (Switched Network) an.

In der Automobilumgebung gelten andere Anforderungen als für Computer Netzwerke in Gebäuden (Störsicherheit, Gewicht, Umwelt-Einflüsse, Echtzeitfähigkeit, Kosten, Länge der Leitungen usw.). Standard-Ethernet-Bauelemente sind wegen des Kabels mit zwei Leitungspaaren und der Steckverbinder-Technik, noch im Hinblick auf die EMV-, Temperatur-, Schüttel- und Feuchtebelastung direkt autotauglich. Um Ethernet automobiltauglich zu machen musste vieles neu entwickelt werden, was die Einführung von Ethernet herausgezögert hat.

Die *OPEN Alliance (One-Pair Ether-Net) Special Interest Group (SIG)*<sup>1</sup> ist ein Interessenverband welcher dass gemeinsame Ziel hat, die Ethernet-Technologie in der Automobilindustrie voranzutreiben und als offenen Standard für automotives Ethernet zu etablieren.

---

<sup>1</sup><http://www.opensig.org/>, Zugriffsdatum: 11.03.2018

## 1. Einleitung

Die OPEN Alliance SIG setzt auf die von Broadcom entwickelte BroadR-Reach® Ethernet Technologie. Diese ermöglicht eine 100 Mbit/s Übertragung über ungeschirmte *Single Twisted Pair Kabel* und bietet dadurch einen deutlichen Kosten- und Gewichts-Vorteil.

Das Ziel dieser Arbeit ist eine ganzheitliche Untersuchung der Ethernet Technologie und deren Bewertung betreffend automobiler Tauglichkeit.

Im theoretischen Teil werden die Grundlagen der Ethernet-Technologie und die am häufigsten eingesetzten Bussysteme im Fahrzeug beschrieben, sowie die Motivation und Stand der Technik von Automotive Ethernet.

Der praktische Teil besteht aus zwei Arbeitspaketen. Im ersten Arbeitspaket wird mit Hilfe von Evaluierungsplattformen ein Demonstrator mit Standard-Ethernet-Bausteinen hard- und softwaremäßig in Betrieb genommen. Dabei wird eine Schnittstelle zwischen CAN und Ethernet realisiert welche Nachrichten untereinander austauschen kann.

Im zweiten Arbeitspaket werden Seriennahe Automotive-Ethernet-Bausteine (BroadR-Reach Transceiver) auf die vorhandene Hardware- und Softwarestruktur adaptiert.

Anschließend werden Untersuchungen (EMV, Signalqualität) hinsichtlich automobiler Tauglichkeit der Ethernet-Technologie durchgeführt.

### 1.1. Aufbau der Arbeit

1. **Literaturrecherche:** Es werden die Grundlagen der Netzwerk-Kommunikation und der Ethernet-Technologie (Abschn. 2), sowie die am häufigsten eingesetzten Bussysteme (Abschn. 4) im Kraftfahrzeug beschrieben. Weiters wird die Motivation (Abschn. 5) hinter der Einführung vom Ethernet ins Auto erläutert und eine Faktensammlung zum Stand der Technik durchgeführt.
2. **Standard-Ethernet Demonstrator:** Im ersten Arbeitspaket (Abschn. 6.2) wird auf Basis zweier *Evaluierungsboards* von Texas Instruments (TI) eine Schnittstelle zwischen CAN (*Controller Area Network*) und Ethernet hergestellt. Diese Schnittstelle soll Nachrichten vom CAN Modul empfangen, auf ein Ethernet taugliches Protokoll (UDP/TCP) umsetzen und über den Ethernet-Transceiver senden. Die Empfängerplatine arbeitet den Signalfluss in der umgekehrten Richtung nochmals ab.
3. **Automotive-Ethernet Demonstrator:** Die physikalische Schicht der Ethernet Schnittstelle wird im zweiten Arbeitspaket durch automobil-taugliche Ethernet-Transceiver ersetzt (Abschn. 6.3). Dafür werden zwei auf der BroadR-Reach® Technologie basierten Platinen entworfen. Anschließend werden Untersuchungen (EMV, Signalqualität) hinsichtlich automobiler Tauglichkeit der Ethernet-Technologie durchgeführt.

## 2. Grundlagen

*In diesem Kapitel werden die Grundlagen der modernen Netzwerkgeräte-Kommunikation und der Ethernet-Technologie beschrieben.*

### 2.1. OSI-Referenzmodell

Wegen der großen Komplexität der Computer-Kommunikationsnetzwerke, wurde die Kommunikation mittels *Schichtenmodellen* unterteilt. Jede Schicht (*Layer*) hat eine bestimmte Aufgabe zur erfüllen, und bietet Schnittstellen zur darüberliegenden und darunterliegenden Schicht. In den jeweiligen Schichten werden die Daten gekapselt (siehe Abb. 2.2). Die bekanntesten Schichtenmodelle sind das TCP/IP-Referenzmodell, das OSI-Referenzmodell und das hybride-Referenzmodell[[1]].

Die *International Organization for Standardization* (ISO) hat das *Open Systems Interconnection* (OSI)-Referenzmodell entwickelt und im Jahre 1983 standardisiert. Das OSI-Referenzmodell basiert auf dem *TCP/IP-Referenzmodell*, auch *DoD-Schichtenmodell* genannt, welches als die Basis des Internets gilt.

Das TCP/IP- und OSI-Referenzmodell sind sehr ähnlich und unterscheiden sich in zwei Punkten. Die Netzzugangsschicht ist beim OSI-Referenzmodell auf Bitübertragungs- und Sicherungs-Schicht aufgeteilt, und die Anwendungsschicht auf Sitzungsschicht, Darstellungsschicht und Anwendungsschicht (siehe Abb. 2.1).

Der Unterschied zwischen dem OSI- und hybriden-Referenzmodell ist, dass die Aufgaben der Anwendungsschicht beim OSI auf die zusätzlichen Schichten, Sitzungsschicht und Darstellungsschicht aufgeteilt sind (siehe Abb. 2.1). Das hybride-Referenzmodell kombiniert die Vorteile des OSI- und TCP/IP-Referenzmodells, ohne deren Nachteile zu übernehmen[[1]].

Das OSI-Referenzmodell definiert sieben aufeinander folgende Schichten, die jeweils Teilfunktionen realisieren und weitestgehend unabhängig voneinander sind. Jede Schicht enthält *Protokolle* (siehe Tab. 2.1), welche die Regeln für eine Kommunikation zwischen zwei Systemen festlegen, und *Diensten* welche aus einer Sammlung von *Operationen* bestehen. Da manche Protokolle für ganz bestimmte Anwendungen entwickelt wurden, kommt es auch vor, dass sich Protokolle über mehrere Schichten erstrecken und mehrere Aufgaben abdecken.

## 2. Grundlagen

Einer Nachricht wird in jeder Schicht ein *Header* hinzugefügt, welche zusätzliche Informationen enthält. Bei einer Kommunikation zwischen zwei Systemen durchläuft die Kommunikation alle sieben Schichten zweimal. Einmal beim Sender und einmal bei Empfänger[[20]].

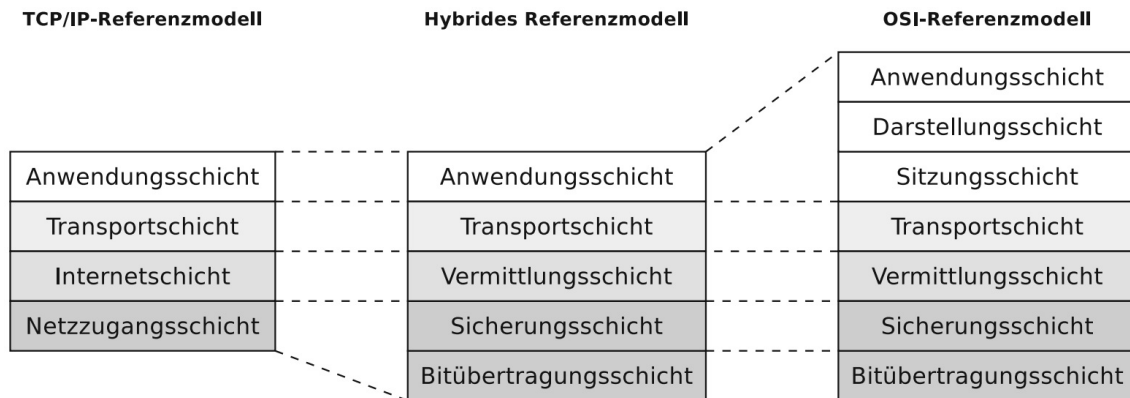


Abbildung 2.1.: Vergleich des TCP/IP, Hybriden- und OSI-Referenzmodells [[1]]

Beschreibung der Schichten des OSI-Referenzmodells (siehe Abb. 2.1):

1. **Bitübertragungsschicht:** Die Aufgabe der Bitübertragungsschicht (*Physical Layer*) ist die Übertragung der einzelnen Bits (Einsen und Nullen) einer Nachricht von einem Netzwerkgerät zum anderen. In dieser Schicht findet die physische Verbindung an das Medium und die Konvertierung der digitalen- in analoge-Signale statt. Die Schicht 1 definiert z.B. die Übertragungsrate, die Belegung der Pins, den Steckertyp, Sende- und Empfangspegel. Unter Hardware der Schicht 1 fallen Hubs, Modems und Repeater[[20]].
2. **Sicherungsschicht:** Eine der Aufgaben der Sicherungsschicht (*Data Link Layer*) ist die Erkennung von Fehlern bei der Übertragung von Bitfolgen. Außerdem regeln die Protokolle der Schicht 2, mit dem Mediumzugriffsverfahren CSMA/CD den Zugriff auf das Medium. Die Sicherungsschicht beim Sender verpackt die Pakete der Vermittlungsschicht in Rahmen (*Frames*) und überträgt diese zum Empfänger, bei welchem die Sicherungsschicht die Rahmen im Bitstrom des Physical Layer erkennt. Für eine Sendung der Frames auf einem physischen Netzwerk sind MAC-Adressen nötig. Die Protokolle der Sicherungsschicht fügen jedem Rahmen eine Prüfsumme zu, welche dem Empfänger ermöglicht fehlerhafte Rahmen zu erkennen. Auf der Sicherungsschicht werden Switches und Bridges eingesetzt[[1]].
3. **Vermittlungsschicht:** Die Vermittlungsschicht (*Network Layer*), im TCP/IP Referenzmodell auch *Internet Layer* genannt, hat die Aufgabe Daten (*Pakete*) in einem logischen Netz vom Sender zum Empfänger zu vermitteln. Dazu definiert der Network-

Layer logische Adressen (IP Adressen). Beim Sender werden die Segmente der Transportschicht von der Vermittlungsschicht in Pakete verpackt, welche beim Empfänger von der Vermittlungsschicht erkannt werden. Die wichtigste Aufgabe dieser Schicht ist *Routing*, welches die Pakete vom Sender zum Empfänger weiterleitet. In der Regel wird das verbindungslose *Internet Protocol* (IP) verwendet. Das IP-Paket wird an das Ziel *geroutet*, wobei der Pfad nicht gespeichert wird[[1]].

4. **Transportschicht:** Die vierte Schicht (*Transport Layer*) übernimmt alle Aufgaben die für den Beginn, Überwachung und Schließung einer Verbindung zwischen zwei Netzwerkgeräten notwendig sind. Dazu zählen z.B. die Segmentierung von Datenpaketen und die Stauvermeidung (*congestion control*). Die Transportschicht sorgt dafür, dass eine vollständige Nachricht in der korrekten Reihenfolge beim Empfänger ankommt. Der Transport-Layer ist nur in den Endsystemen und nicht in den Netzknoten implementiert. D.h. eine Anwendung an einer Quellmaschine führt ein Gespräch mit einer ähnlichen Anwendung auf der Zielmaschine, die durch eine Vielzahl von Netzknoten getrennt sein können[[32]]. Beispiele für Protokolle der Transportschicht sind das verbindungslose UDP (User Datagram Protocol) (Abschn. 2.1.1.1) und das verbindungsorientierte TCP (Transport Control Protocol) (Abschn. 2.1.1.2)[[20]].
5. **Sitzungsschicht:** Die Sitzungsschicht (*Session Layer*) übernimmt die Aufgaben, die notwendig sind um zwei Anwendungen in einem Netzwerk zu verbinden. Das sind Beginn, Überwachung und Abschluss einer Sitzung. Dienste des Session-Layer sind *Synchronisation* und *Dialogsteuerung*. Bei der Synchronisation werden Markierungen in den Datenstrom gelegt, sodass nach einer Unterbrechung nur die Daten ab der letzten Markierung neu verschickt werden. Die Dialogsteuerung entscheidet bei einer Sitzung im Halbduplex-Modus, welcher Client an die Reihe kommt.[[20]]
6. **Darstellungsschicht:** In der Darstellungsschicht (*Presentation Layer*) befinden sich Protokolle zur Formatierung der Nachrichten. Wenn eine Nachricht in einem speziellen Format vorliegt (z.B. ASCII), kann der Sender den Empfänger darüber informieren und so die eventuell nötige Konvertierung beim Empfänger zu ermöglichen[[1]].
7. **Anwendungsschicht:** Die Anwendungsschicht (*Application Layer*) beinhaltet verschiedene Anwendungsprogramme wie z.B. http (Internet), SMTP (E-Mail), telnet (Terminal) und ftp (Datei-Transfer)[[20]].

Die Bitübertragungsschicht und Sicherungsschicht wird im Abschn. 2.2 detaillierter beschrieben.

## 2. Grundlagen

Der Ablauf der Datenkapselung wird in der Abb. 2.2 am Beispiel des hybriden-Referenzmodells beschrieben.

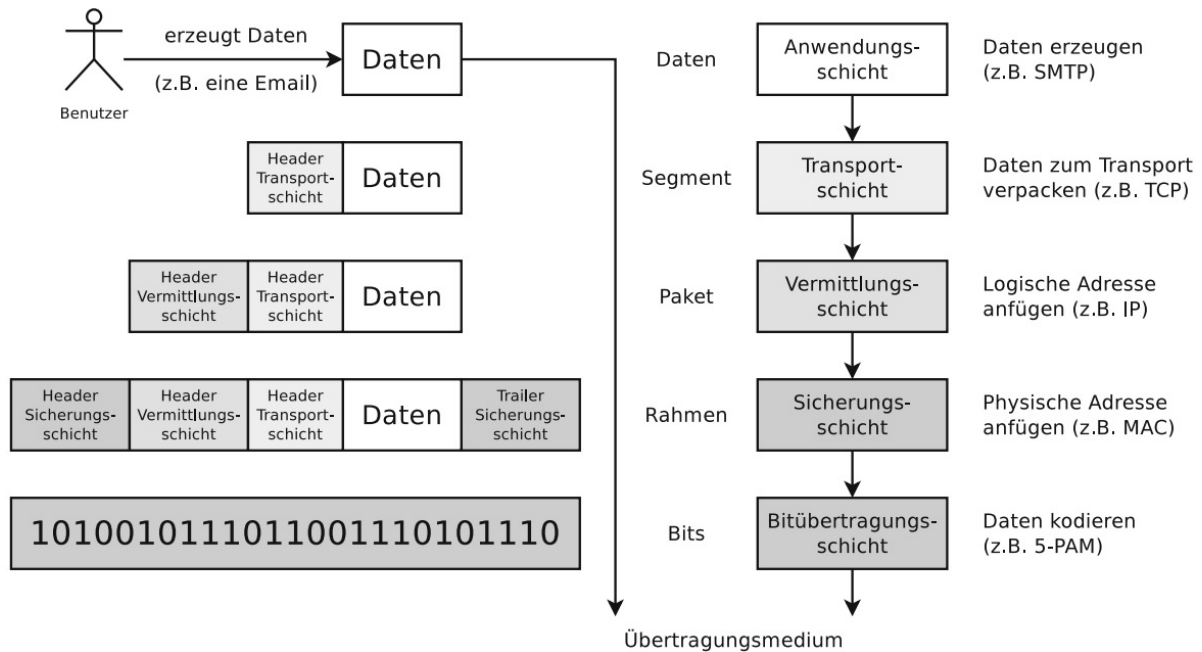


Abbildung 2.2.: Datenkapselung am Beispiel des hybriden-Referenzmodells bei vertikaler Kommunikation[[1]]

Bei der *vertikalen-Kommunikation* durchlaufen Daten die Schichten des eingesetzten Referenzmodells. Die Nachricht wird beginnend von der oberen Schicht nach unten bei jeder Schicht neu verpackt.

Beim Empfänger wird die Nachricht in umgekehrter Richtung von der unteren Schicht zur oberen immer weiter entpackt. Beim Sender werden den Daten ein *Header* und *Trailer* dazugegeben, welche beim Empfänger in den jeweiligen Schichten ausgewertet und anschließend verworfen werden.

Diese beschriebenen Vorgänge heißen Datenkapselung (*Data Encapsulation*) und Entkapselung (*De-Encapsulation*)[[1]].



In der folgenden Tabelle 2.1 werden die verschiedenen Protokolle den Schichten des OSI-Referenzmodells zugeordnet. Das Übertragungsmedium, welches nicht Bestandteil des OSI-Modells ist, wird gelegentlich als *Schicht 0* bezeichnet[[20]].

Tabelle 2.1.: Übersicht der Protokolle zugeordnet in dazugehörige Schichten des OSI-Referenzmodells[[20]]

Netzabschnitt OSI-Layer	Teilnehmernetz/ Endgerät	Teilnehmer- Anschluss- Leitung	Backhaul	Backbone
Layer 7	http, ftp, telnet, SMTP etc.	N/A	N/A	N/A
Layer 6		N/A	N/A	N/A
Layer 5		N/A	N/A	N/A
Layer 4	TCP, UDP	N/A	N/A	N/A
Layer 3	IP, ISDN	N/A	N/A	IP, ISDN, POTS, (ATM)
Layer 2	Ethernet, FDDI, Token Ring	ATM, Ethernet, PPPoE	ATM, Ethernet, PPPoE, RPR, GFP	PPP/HDLC, Ethernet, ATM, FR, GFP
Layer 1	POTS, ISDN, Ethernet, FDDI, Token Ring, DECT, WLAN, PLC, Bluetooth, ZigBee	POTS, ISDN, xDSL, FTTx, Ethernet (EFM) WIMAX, PMP, GSM, UMTS	PDH, SDH, WDM (CWDM), Ethernet	PDH, SDH, WDM (DWDM), OTH
„Layer 0“	Kupferdoppel- ader, LWL, Funk, Koaxial- kabel, Strom- leitung	Kupferdoppel- ader, LWL, Funk, Koaxial- kabel, Strom- leitung	Lichtwellenleiter, Richtfunk	Lichtwellenleiter (LWL)

## 2. Grundlagen

### 2.1.1. Transport Layer

In den ersten drei Schichten des OSI-Referenzmodells befinden sich Technologien die ermöglichen, Netzgeräte miteinander zu verbinden und die Daten auszutauschen. Die Transportschicht enthält Protokolle (Tab. 2.1) für die Kommunikation zwischen Anwendungen des Quell- und Ziel-Netzwerkgerätes. Die Daten der Anwendungsschicht werden in der Transportschicht in *Segmente* verpackt und die Anwendungen mit *Portnummern* adressiert[[1]].

Anwendungen die TCP oder UDP nutzen besitzen eine *Portnummer*, die angibt welche Anwendungen am Zielgerät angesprochen werden. Die Portnummer ist 16 Bits groß (0 bis 65535) und kann in der Regel beliebig vergeben werden. Es gibt aber Standardanwendungen die einen fixen Port besitzen. Ein *Socket* ist eine plattformunabhängige, standardisierte Schnittstelle zwischen den OS-Netzwerkprotokollen und den Anwendungen. Dieser besteht aus einer Portnummer und einer IP-Adresse.

*Datagram-Sockets* verwenden das verbindungslose UDP-Protokoll , wobei *Stream Sockets* das verbindungsorientierte TCP-Protokoll verwenden[[1]]. Beide Protokolle wurden in dem praktischen Teil der MA verwendet und werden in der weiteren Folge beschrieben.

#### 2.1.1.1. UDP

Das *User Datagram Protocol* (UDP) überträgt Daten ohne vorher eine Verbindung aufgebaut zu haben und ist ausschließlich für die Adressierung der Segmente zuständig. Die empfangenen Nachrichten werden nicht vom Empfänger beim Sender bestätigt und deshalb können einzelne Segmente verloren gehen.

Im Gegensatz dazu wird bei Videostreaming via TCP ein Wiedergabepuffer angelegt, da ein verlorenes Segment neu angefordert wird und es so zu Aussetzern kommt. Bei Verwendung vom UDP (bei z.B. Videostreaming oder Videotelefonie) geht nur ein Bild verloren beim Nichteintreffen von einem Segment.

Ein UDP-Segment hat eine maximale Größe von 65535 Bytes und wird vom IP fragmentiert übertragen. Die Abb. 2.3 zeigt den UDP-Header, welcher aus vier 16 Bit-Datenfeldern besteht[[1]].

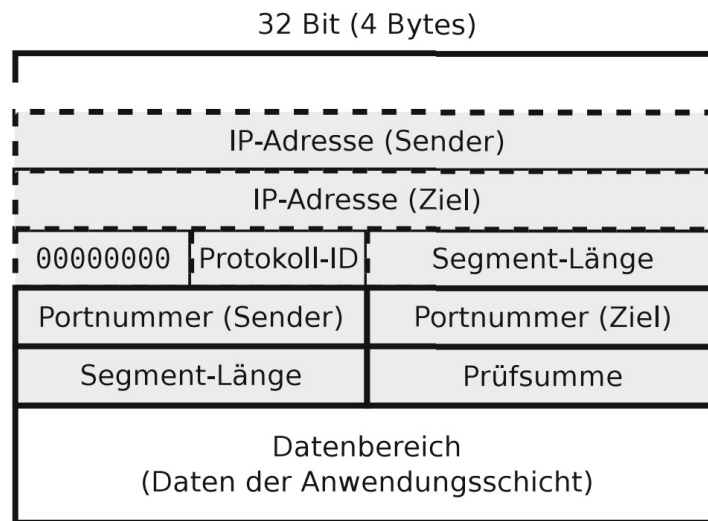


Abbildung 2.3.: UDP-Segments inkl. Pseudo-Header[[1]]

Im ersten Datenfeld befindet sich die Portnummer des Prozesses der sendet, welche dem Empfänger sagt wem er antworten muss. Im zweiten Feld steht die Portnummer des Prozesses, für welchen das Segment bestimmt ist. Die zwei letzten Felder enthalten die Länge des Segments und eine Prüfsumme.

Es wird ein Pseudo-Header erzeugt (Abb. 2.3) der Informationen des Network Layer z.B. IP-Adressen vom Sender und Empfänger enthält[[1]].

### 2.1.1.2. TCP

Das *Transmission Control Protocol* (TCP) ist ein verbindungsorientiertes Protokoll, welches die notwendige Zuverlässigkeit bei ausgewählten Anwendungen bietet. Das heißt, bei TCP ist garantiert, dass vollständige Segmente und in der korrekten Reihenfolge beim Empfänger ankommen. Bei einem verlorenen oder unbestätigten Segment sendet es der Sender erneut.

Bei TCP werden die Nutzdaten als geordneter Datenstrom angesehen, welche mit *Sequenznummern* (Folgenummern) nummeriert werden. Die Sequenznummer eines Segments zeigt auf die Position des ersten Bytes des Segments im Datenstrom. Diese Sequenznummern ermöglichen dem Empfänger bei Bedarf, die Reihenfolge der Segmente zu korrigieren und doppelt empfangene Segmente zu verwerfen.

Durch die angegebene Länge des Segments im IP-Header können Lücken im Datenstrom entdeckt werden und so verlorene Segmente vom Empfänger neu angefordert werden.

## 2. Grundlagen

Die Abb. 2.4 zeigt das TCP-Segment inklusive dem Pseudo-Header[[1]].

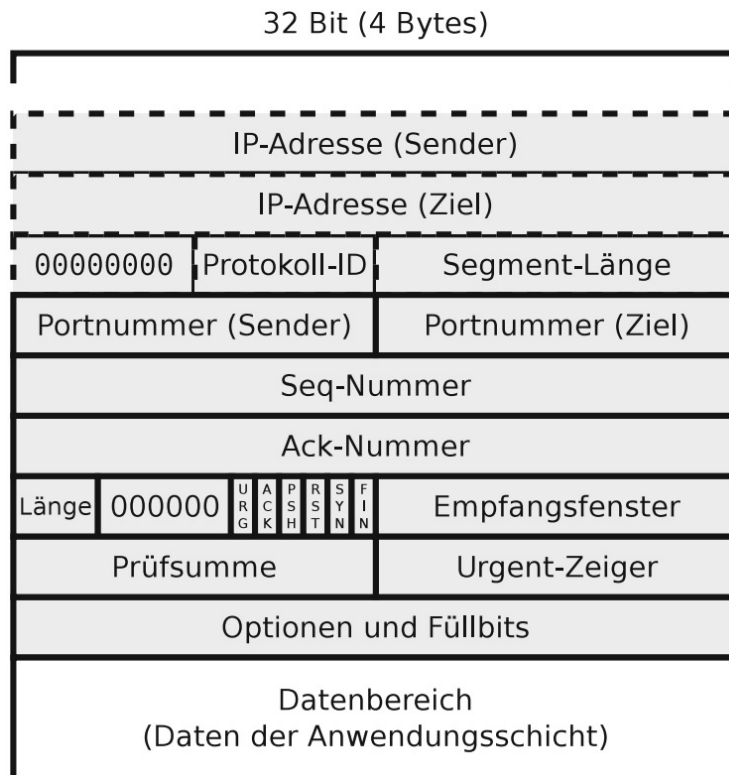


Abbildung 2.4.: TCP-Segments inkl. Pseudo-Header[[1]]

Ein TCP-Segment hat eine maximale Größe von 65535 Bytes, wobei dieses bei Ethernet üblicherweise weniger als 1500 Bytes beträgt. Der TCP-Header ist um einiges komplexer als der bei UDP und hat eine Größe von 20 Byte.

Je ein Datenfeld enthält die Portnummer des Prozesses das sendet und die Portnummer des Prozesses für den das Segment bestimmt ist. Die Sequenznummer (*Seq-Nummer*) enthält die Nummer des aktuellen Segments und *Ack-Nummer* die Nummer des nachfolgenden Segments. Das Feld *Länge* ist 4 Bits groß. Das nächste Feld ist 6 Bit groß und wird nicht verwendet. Es folgen sechs 1 Bit große Felder, die für den Datenaustausch, Verbindungsaufbau und Verbindungsabbau verwendet werden. Das *Empfangsfenster* enthält die Nummer der freien Bytes beim Sender und dient zur Flusskontrolle[[1]].

Eine TCP-Verbindung wird durch einen *Dreibege-Handshake* aufgebaut. Bei diesem tauschen beide Netzwerkgeräte in drei Schritten Kontrollinformationen aus. So wird gewährleistet, dass der Kommunikationspartner anwesend und empfangsbereit ist. In der Abb. 2.5 wird der TCP-Dreibege-Handshake dargestellt[[1]].

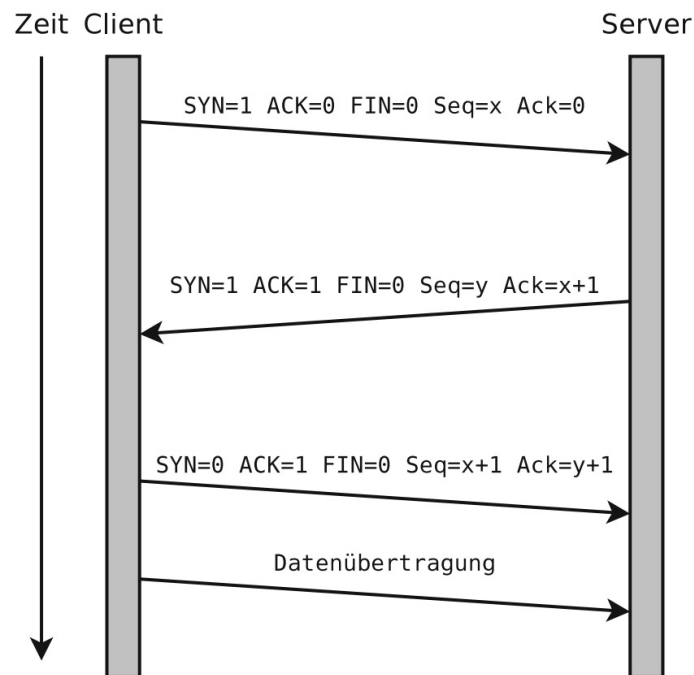


Abbildung 2.5.: TCP-Dreiwege-Handshake[[1]]

1. Im ersten Schritt schickt der *Client* ein Segment zum *Server*, um ihre Sequenznummern zu synchronisieren. Mit dem gesetzten SYN-Bit im Segment, gibt der Client seine Start-Sequenznummer bekannt.
2. Darauf antwortet der Server mit einem Segment, in welchem er mit gesetztem ACK-Bit den Empfang des Client-Segments bestätigt und einem gesetztem SYN-Bit für die Synchronisation der Sequenznummern.
3. Im dritten Schritt antwortet der Client mit einem Segment mit gesetztem ACK-Bit, womit er den Empfang bestätigt. Ab jetzt können Daten zwischen den Kommunikationspartnern ausgetauscht werden[[1]].

## 2. Grundlagen

### 2.2. Ethernet

Ethernet wurde Anfang der 1970er Jahre von Robert Metcalfe im Forschungslabor der Firma Xerox am PARC (Palo Alto Research Center) entwickelt. Das Ziel war es alle Rechner im Gebäude mit einem Laserdrucker zu verbinden. In späterer Zusammenarbeit zwischen den Firmen Xerox, DEC und Intel, wurde Ethernet zu einem offenen Standard.

Das *Institute of Electrical and Electronics Engineers* (IEEE) hat die erste Version von Ethernet ab 1980 in der Arbeitsgruppe 802<sup>1</sup> weiterentwickelt. Ethernet wird im CSMA/CD (802.3), Token Bus (802.4) und dem Token Ring (802.5) Standard beschrieben. Abb. 2.6 zeigt die IEEE 802 LAN- und MAN-Standards[[23]].

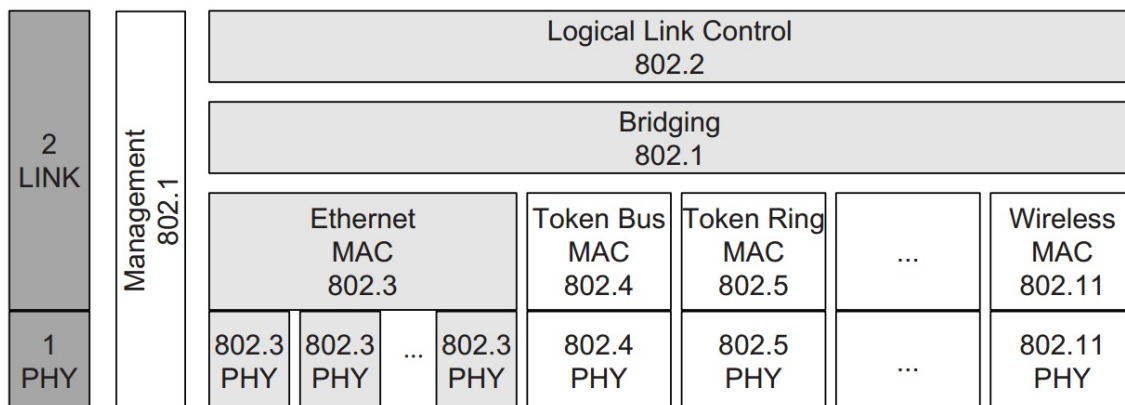


Abbildung 2.6.: IEEE 802 LAN- und MAN-Standards[[23]]

1985 hat IEEE den ersten 802.3 Standard veröffentlicht. In den folgenden Jahren wurde Ethernet ständig erweitert z.B. die Übertragungsraten erhöht und neue Übertragungsmedien hinzugefügt. Außerdem wurden Mechanismen wie *Vollduplex* oder *Autonegotiation* einbezogen. Das alles hat Ethernet verholpen sich gegenüber den konkurrierenden LAN-Technologien durchzusetzen[[1]]. Zurzeit sind weltweit etwa 85 % aller PCs und *Workstations* in einer LAN-Umgebung über Ethernet miteinander verbunden[[11]].

Ethernet deckt die Bitübertragungsschicht (Physical Layer) und Sicherungsschicht (Data Link Layer) des OSI-Referenzmodells (siehe Abb. 2.7)[[20]] ab, und definiert die Zugriffskontrolle und Adressierung auf das Übertragungsmedium.

<sup>1</sup><http://www.ieee802.org/>, Zugriffsdatum: 11.03.2018

Der Ethernet Physical Layer (PHY) besteht bei Übertragungsraten bis 10 MBit/s aus *Physical Layer Signaling* (PLS) und *Physical Medium Attachment* (PMA) Layer. Bei Übertragungsraten von 100 MBit/s und mehr besteht dieser aus dem Sublayern: *Physical Coding Sublayer* (PCS), PMA und *Physical Medium Dependent* (PMD) (siehe Abb. 2.7)[[20]].

*Medium Independent Interface* (MII) ist die Schnittstelle zwischen dem MAC- und dem PHY-Layer. Beim 10 MBit/s Ethernet und Fast Ethernet heißt diese Schnittstelle MII. Für Übertragungsraten über 100 Mbit/s verwendet man den xMII Namen. Bei 1 Gbit/s Ethernet GMII und bei 10 Gbit/s XGMII[[13]]. Aus physikalischen Gründen ist die Verwendung von abgesetzten PLS-Baugruppen bei Übertragungsraten von 1000 Mbit/s und höher nicht möglich. Die GMII und XGMII bezeichnen Schnittstellen zwischen einzelnen Chips, im Gegensatz zu Fast Ethernet zwischen Ethernet-Baugruppen[[23]].

Die zweite Schicht des OSI-Referenzmodells (Data Link Layer) besteht aus dem *Logical Link Control* (LLC), dem *MAC Control Layer* und dem *Medium Access Control* (MAC) Layer. Der LLC ist ein Netzprotokoll, welches eingehende Daten an entsprechende Protokolle des Network-Layers weiterleitet. Daten von dem Network-Layer werden von LLC an den MAC weitergegeben. LLC bietet einen Verbindungslosen Dienst (LLC 1), einen Verbindungsorientierten Dienst (LLC 2) und einen bestätigten Dienst (LLC 3). Die MAC-Schicht definiert die Zugriffssteuerung der Stationen auf den gemeinsamen Physical Layer[[20]]. Die beiden Sublayer sind unabhängig vom Ethernet-Standard und werden auch bei anderen Netzwerktechniken verwendet. Die Abb. 2.7 zeigt die Einordnung der Ethernet-Schnittstellen in das OSI-Referenzmodell.

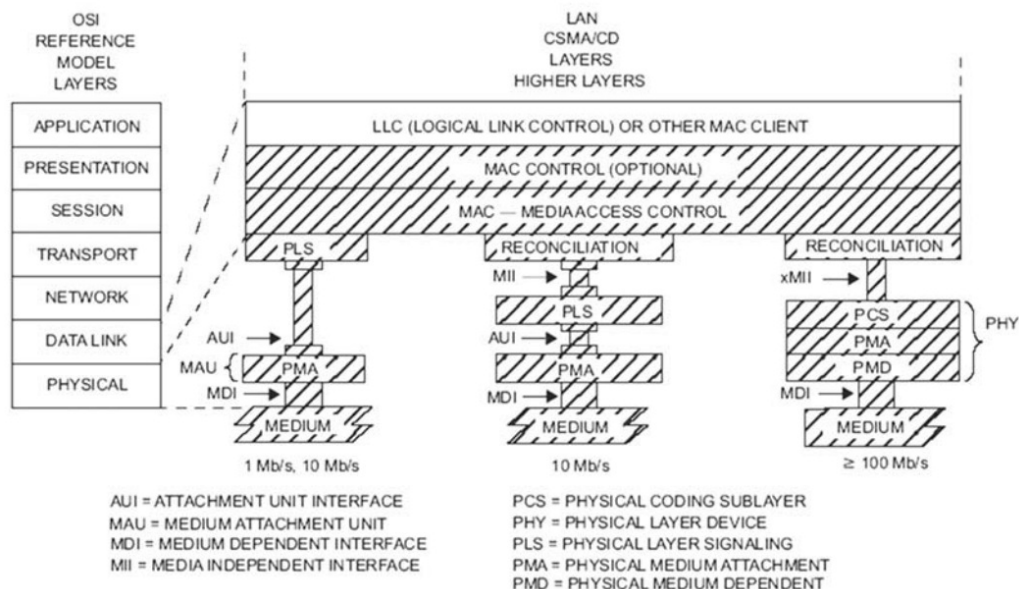


Abbildung 2.7.: Einordnung der Ethernet-Netzarchitektur in das OSI-Referenzmodell[[13]]

## 2. Grundlagen

### 2.2.1. Standards

Der IEEE 802.3 Standard umfasst eine ganze Reihe von Techniken, welche einzeln standardisiert sind. Diese Ethernet-Standards unterscheiden sich in der Übertragungsrate, Reichweite, Leitungscodierung und dem Übertragungsmedium.

Bislang sind Übertragungsraten von 10 MBit/s, 100 MBit/s (Fast Ethernet), 1000 MBit/s (Gigabit Ethernet) und bis maximal 10 GBit/s spezifiziert. Als Übertragungsmedium werden Koaxial-Kabel, Twisted-Pair-Kabel oder Glasfaser-Kabel verwendet[[1]].

#### 2.2.1.1. 10 Mbit/s Ethernet

Ethernet wurde mit dem Ziel entwickelt, mehrere Netzwerkgeräte (z.B. PCs, Server, Drucker) miteinander zu verbinden und deren Datenaustausch zu ermöglichen. Alle Stationen werden in einer Bus-Topologie aufgebaut, wo sich mehrere Stationen das Übertragungsmedium teilen. Die Übertragungsrate auf dem Kanal betrug 10 MBit/s[[20]].

Ethernet deckt den Physical Layer und den Data Link Layer des OSI-Referenzmodells ab. Es verwendet eine einfache Manchesterkodierung, die je Datenbit zwei Leitungsbits überträgt, wodurch es zu einer Verdopplung der Signalisierungsrate kommt, währenddessen das Spektrum bis 10 MHz reicht.

Ethernet bereitet Probleme bei allen Arten von zeitkritischen Anwendungen, weil es die Pakete ohne ein festes Zeitraster transportiert. Daher gibt es keine Garantie, dass die Daten innerhalb einer bestimmten Zeit ankommen werden.



Die Tabelle 2.2 gibt einen Überblick über die verschiedenen 10 Mbit/s Ethernet-Schnittstellen.

Tabelle 2.2.: 10 Mbit/s Ethernet-Schnittstellen[[23]]

Schnittstelle	Reichweite	Übertragungsmedium	Leitungscode	Duplex	Anzahl Stationen	Bemerkung
1 BASE-5	250 m 4 km <sup>1</sup>	zwei CuDA (UTP, Telefonkabel)	Manchester		2 (pt-pt <sup>2</sup> )	
2 BASE-TL	2,7 km	eine oder mehrere CuDA (Telefonkabel)	64B/65B DMT	full	2 (pt-pt)	EFM
10 BASE-2 (Cheapernet)	185 m	50 Ω Koax (RG58)	Manchester	half	30 (Bus)	
10 BASE-5	500 m 2,5 km <sup>3</sup>	50 Ω Koax (RG213)	Manchester	half	100 (Bus)	
10 BASE-T	100 m	CuDA 100 Ω (UTP-3)	Manchester	half/ full	2 (pt-pt)	
10 BASE- FP	1 km <sup>4</sup>	MM-Glasfaser 62,5/125 μm λ = 850 nm	Manchester	half/ full	33 (passiver optischer Stern)	P: passive
10 BASE-FL	2 km	MM-Glasfaser 62,5/125 μm λ = 850 nm	Manchester	half/ full	2 (pt-pt)	L: link
10 BASE-FB	2 km	MM-Glasfaser 62,5/125 μm λ = 850 nm	Manchester	half	2 (pt-pt)	B: Backbone
10 BROAD-36	2,8 km <sup>5</sup>	ein oder mehrere 75 Ω Koax CATV-Kabel	PSK	half	Bus	
10 PASS-TS	750 m	ein oder mehrere CuDA (Telefonkabel)	64B/65B DMT		passiver optischer Stern	EFM

Zunächst wurden nur Koaxial-Kabel (RG58, RG213) als Übertragungsmedium verwendet. Später kamen die heute üblichen Twisted-Pair- und Glasfaser-Kabel hinzu. Diese Schnittstellen werden in einer Punkt-zu-Punkt Konfiguration betrieben, wobei eine Halb- und Vollduplex-Verbindung möglich ist[[20]].

## 2. Grundlagen

Das 10 MBit/s Ethernet hat ursprünglich im Halbduplex-Modus gearbeitet, welches auf dem CSMA/CD Zugriffsverfahren basiert und Koaxial Kabel verwendet. Eine Weiterentwicklung ist der Vollduplex-Modus, der auf das Medienzugriffsverfahren CSMA/CD verzichtet und später auch bei 10 MBit/s Ethernet eingeführt wurde.

Bei Halbduplex kann an einer Schnittstelle zu einem Zeitpunkt gesendet oder empfangen werden, aber nicht beides gleichzeitig. Der Vollduplex-Modus funktioniert nur in einer Point-to-Point Verbindung und verwendet ein Twisted-Pair-Kabel. Vollduplex erlaubt eine gleichzeitige Kommunikation in beide Richtungen, durch die separaten Empfangs- und Sendepfade. Alle weiterentwickelten Ethernet-Standards arbeiten im Vollduplex Modus. Die Abb. 2.8 beschreibt den Voll- und Halbduplex-Modus.

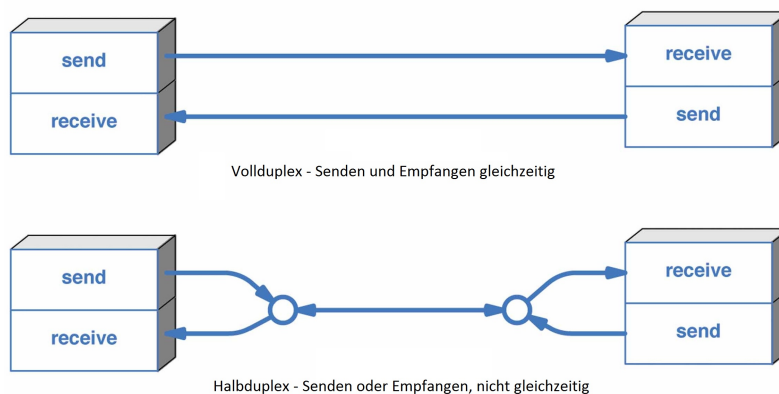


Abbildung 2.8.: Voll- und Halbduplex Modus

### 2.2.1.1.1. CSMA/CD

Mit dem *Carrier Sense Multiple Access/Collision Detection* (CSMA/CD) Medienzugriffsverfahren bei Ethernet wird der Zugriff der Netzwerkgeräte auf das Übertragungsmedium geregelt.

*Carrier Sense* (CS) heißt, dass jedes Netzwerkgerät den Übertragungskanal ständig überwacht, und nur dann senden darf, wenn der Kanal frei ist. *Multiple Access* (MA) sagt aus, dass alle Netzwerkgeräte auf das gleiche Übertragungsmedium zugreifen wollen. *Collision Detection* (CD) heißt, dass jedes Netzwerkgerät auch während es Daten sendet den Kanal überwacht, um Kollisionen zu vermeiden und eine Fehlerbehandlung durchführen[[1]].

Im Falle das zwei oder mehrere Netzwerkgeräte gleichzeitig auf das Übertragungsmedium zugreifen, kommt es zu einer Kollision. Wird eine Kollision von den beteiligten Netzwerkgeräten erkannt, senden diese ein spezielles Signal (Jam-Signal) an alle Teilnehmer, damit alle die Kollision erkennen können.

Das Netzwerkgerät erkennt eine Kollision über den Vergleich des gesendeten und empfangenen Signals. Stimmen die beiden Signale nicht überein, so weiß das Netzwerkgerät dass es zu einer Kollision gekommen ist. Die Sender warten dann eine zufällige Zeit gemäß einem speziellen Algorithmus (exponentieller Backoff-Algorithmus). Danach wird wieder überprüft ob der Bus frei ist, und wenn ja, dann beginnen sie erneut zu senden. Es gibt drei Zustände: der Bus kann frei sein, belegt sein oder es hat eine Kollision stattgefunden[[32]][[1]].

Die Abb. 2.9 zeigt das Flussdiagramm des Sendevorgangs aus der Sicht eines Netzwerkgerätes.

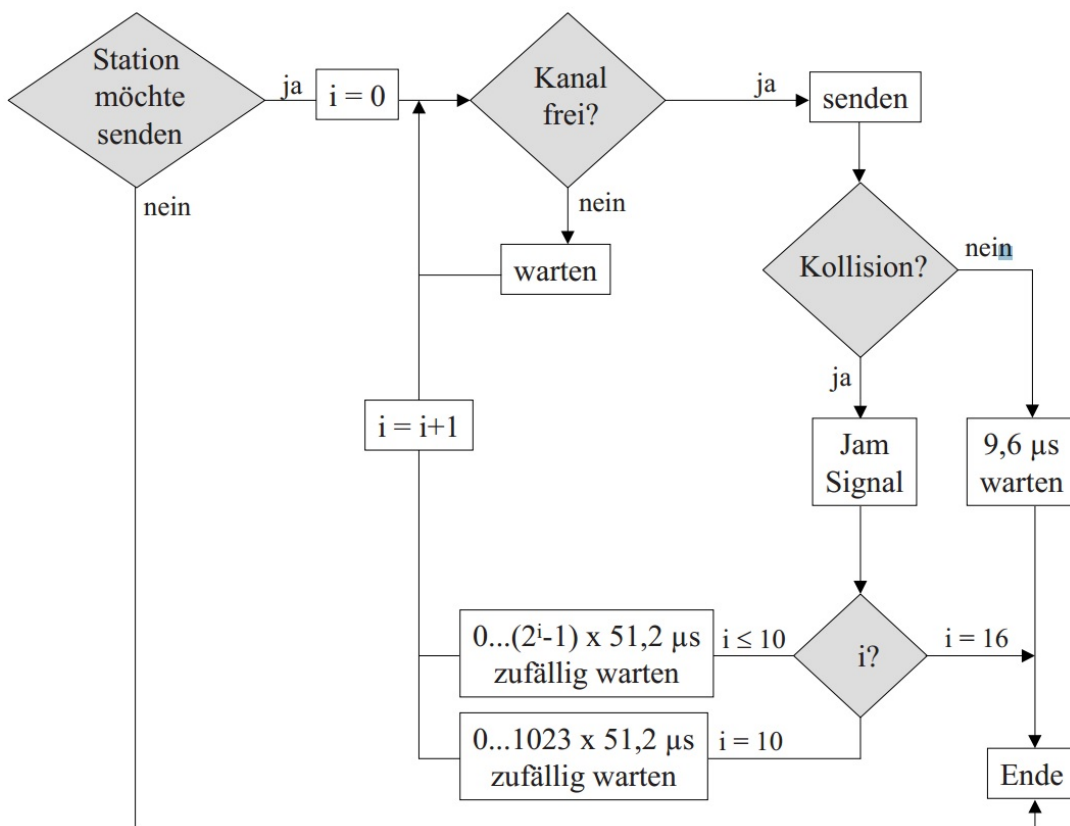


Abbildung 2.9.: Flussdiagramm des Sendevorgangs aus der Sicht eines Netzwerkgerätes[[1]]

Im Halbduplex-Modus gehören Kollisionen zum Normalbetrieb dazu. Die Kollisionsdomäne beschreibt einen Teilbereich eines Netzwerks, bestehend aus Netzwerkgeräten der ersten Schicht des OSI-Referenzmodells.

Beim Switch gibt es keine Kollisionsdomänen mehr, da jede Verbindung zwischen dem Switch und einem Netzwerkgerät eine eigene Kollisionsdomäne bildet. Nur beim Einsatz von Hubs müssen diese beachtet werden.

## 2. Grundlagen

### 2.2.1.1.2. Ethernet Rahmen

Die Abb. 2.10 zeigt den IEEE 802.3 Ethernet-Rahmen mit VLAN-Tag. Ein Ethernet-Rahmen ist inklusive Präambel und *Start Frame Delimiter* (SFD) minimal 72 Bytes und maximal 1526 Bytes groß. Dem Rahmen wird eine Präambel vorangestellt, die aus einer 7 Byte langen Bitfolge: *10101010* besteht und die bei Bus-Topologien zur Synchronisation der Empfänger dient. Darauf folgt SFD der 1 Byte groß ist (Bitfolge: *10101011*), welcher den Beginn des Ethernet-Rahmens anzeigt[[20]].

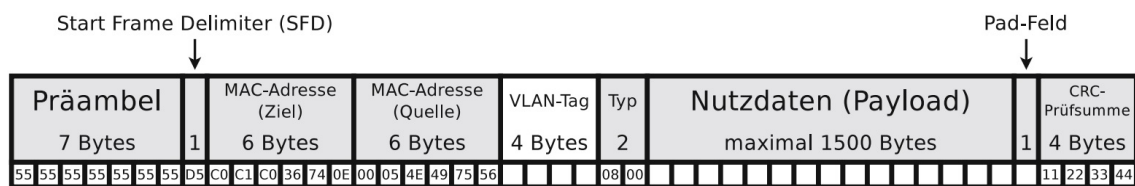


Abbildung 2.10.: IEEE 802.3 Ethernet-Rahmen mit VLAN-Tag[[1]]

Der Ethernet-Rahmen beginnt mit der Ziel-MAC-Adresse (6 Byte), gefolgt von der ebenso 6 Byte langen Quelle-MAC-Adresse. Die MAC-Adressen dienen zur Identifizierung der Netzwerk-Geräte, und sind deren Hardware-Adressen, die vom Hersteller einmalig für jeden vergeben werden.

Der VLAN-Tag ist optional und hat eine Länge von 4 Byte. Er enthält eine 12 Bit lange VLAN-ID und ein 3 Bit großes Priorisierungs-Feld. Das Segment *Typ* beinhaltet das verwendete Protokoll der höheren Schicht, welches bei IPv4 den Wert *0x0800* und bei IPv6 den Wert *0x86DD* hat.

Ein Ethernet-Rahmen kann maximal 1500 Bytes Nutzdaten (Payload) enthalten und muss eine Mindestlänge von 72 Byte haben. Wenn die Nutzdaten weniger als 72 Bytes betragen, wird der Rest automatisch aufgefüllt. Bei Bedarf wird das Ethernet-Frame mit dem *Padding-Bits* (PAD)-Feld auf die Minimalgröße gebracht und so die Kollisionserkennung mit CSMA/CD gewährleistetet[[1]].

Das Ethernet-Frame endet mit einer 4 Byte langen CRC-Prüfsumme, die über das gesamte Frame berechnet wird. Die Prüfsumme geht von der Ziel-MAC-Adresse bis zum PAD-Feld und schließt die Präambel und den SFD aus.

## 2.2.1.2. Fast Ethernet

Fast Ethernet ist eine Weiterentwicklung des *10BASE-T* (802.3i) Ethernet-Standards und wurde 1995 von der IEEE unter den Namen 802.3u standardisiert. Die Übertragungsrate wurde auf 100 Mbit/s erhöht und Übertragungsmedien: Twisted-Pair-Kabel und Glasfaser-Kabel eingeführt.

Alle anderen in Abschn. 2.2.1.1 beschriebenen Eigenschaften von Ethernet sowie der Ethernet-Rahmen (siehe Abschn. 2.2.1.1.2) wurden bei Fast Ethernet beibehalten[[20]]. Die Tabelle 2.3 gibt einen Überblick über die verschiedenen 100 Mbit/s Ethernet-Schnittstellen.

Tabelle 2.3.: Fast Ethernet-Schnittstellen[[23]]

Schnittstelle	Reichweite	Übertragungsmedium	Leitungscode	Duplex	Anzahl Stationen	Bemerkung
100 BASE-T2	100 m 200 m <sup>1</sup>	zwei CuDA 100 Ω (UTP-3/4/5)	QAM 25	full	2 (pt-pt)	Symbolrate 25 Mbaud
100 BASE-T4	100 m 200 m <sup>1</sup>	vier CuDA 100 Ω (UTP-3/4/5)	8B6T	half	2 (pt-pt)	
100 BASE-TX	100 m 200 m <sup>2</sup> 300 m <sup>3</sup>	zwei CuDA (UTP-5 oder 150 Ω STP)	4B5B <sup>4</sup>	full	2 (pt-pt)	
100 BASE-FX	400 m <sup>5</sup> 2 km <sup>6</sup>	zwei MM-Glasfasern	4B5B <sup>4</sup>	half/ full	2 (pt-pt)	
100 BASE-LX10	10 km	zwei SM-Glasfasern $\lambda = 1310 \text{ nm}$				EFM
100 BASE-BX10	10 km	1 SM-Glasfaser $\lambda = 1310 \text{ nm}/$ 1550 nm				EFM

SM: Singlemode-Glasfaser  
 STP: Shielded Twisted Pair  
 QAM: Quadratur Amplituden Modulation

1 maximal 2 Repeater  
 2 mit einem Repeater  
 3 mit 2 Repeatern  
 4 wie bei FDDI (Fiber Distributed Data Interface)  
 5 Halbduplex  
 6 Vollduplex

Bei Fast Ethernet werden Stationen im Gegensatz zum 10 Mbit/s Ethernet über einen Switch oder Hub sternförmig miteinander verbunden. Der Hub übernimmt die Aufgaben vom Repeater.

## 2. Grundlagen

Dieser hat mehrere Ports und wird auch als Multiport-Repeater bezeichnet. Verschiedene 100 Mbit/s Ethernet-Schnittstellen, die über einen Hub miteinander verbunden werden, bilden eine Kollisionsdomäne[[20]].

Zwei wichtige Verbesserungen von Fast Ethernet gegenüber 10 MBit/s Ethernet sind die Einführung des Vollduplex-Modus und der Auto-Negotiation. Der Vollduplex Modus wurde nachträglich auch bei 10 MBit/s Ethernet eingeführt und wird ausschließlich für *Point-to-Point* (Punkt-zu-Punkt) Verbindungen verwendet.

Mit *Auto-Negotiation* können alle wichtigen Verbindungsparameter (Übertragungsrate, Halb- oder Vollduplex, symmetrische oder asymmetrische Flusskontrolle) zwischen zwei Stationen automatisch ausgemacht werden[[20]].

Fast Ethernet arbeitet meistens im Vollduplex Modus, bei welchem CSMA/CD nicht eingesetzt wird und daher eine sogenannte *Flow Control* (Flusskontrolle) erforderlich ist. Die Flusskontrolle stellt sicher, dass Daten nur dann gesendet werden wenn der Empfänger auch empfangsbereit ist. Mit der Flusskontrolle und der Point-to-Point Verbindung gehören Kollisionen und Kollisionsdomänen bei Ethernet der Vergangenheit an.

Die Fast Ethernet Schnittstelle 100BASE-TX ist heutzutage die Standard Ethernet Implementation. Wie beim 10BASE-T benutzt diese ein Verdrilltes-Adernpaar (Twisted Pair) pro Richtung und braucht mindestens ein *Unshielded Twisted Pair* (UTP) Kabel der Kategorie 5. Es werden keine Koaxialkabel mehr verwendet. Statt der Manchesterkodierung in 10 Mbit/s Ethernet wird der effizientere 4B5B-Leitungscode benutzt.

### 2.2.2. Topologien

Die Topologie eines Rechnernetzes beschreibt die physikalische Anordnung von Teilnehmer-Stationen die über Netzwirkabel miteinander verbunden sind. Außerdem die eingesetzte *Hardware*, sowie die Zugriffsmethode auf das Übertragungsmedium. Die Struktur großer Netzwerke setzt sich oftmals aus mehreren unterschiedlichen Topologien zusammen. Zwei übliche Ethernet-Topologien sind Bus- und Stern-Topologie.

### 2.2.2.1. Bus-Topologie

Im ursprünglichen Ethernet wurde ein Koaxialkabel als Übertragungsmedium genutzt. Die einzelnen Stationen wurden mit jeweils einem Kabel mit mehreren anderen Stationen verbunden. So ein Netzwerk wurde als so genannter *Bus* aufgebaut. Vorteile der Bus-Topologie liegen in der einfachen Verkabelung und Netzwerkerweiterung und dementsprechend geringen Kosten. *Thin Ethernet* (10BASE2) und *Thick Ethernet* (10BASE5) sind Beispiele für Netzwerktechnologien die auf der Bus-Topologie basieren.

Bei der Bus-Topologie kommt es beim Aussetzen einzelner Knoten nicht zum Ausfall des Netzwerks, währenddessen eine Unterbrechung des Buses zum Ausfall des kompletten Netzwerks führt. Zum selben Zeitpunkt kann nur ein Gerät Daten auf den Bus senden, da es sonst zu Kollisionen kommen kann. Um den Zugriff auf das Medium zu regeln, kann ein so genannter *Bus-Arbiter* verwendet werden, oder ein Medienzugriffsverfahren z.B. CSMA/CD. Die Ethernet-Bus-Topologie ist in der Abb. 2.11 dargestellt[[1]].

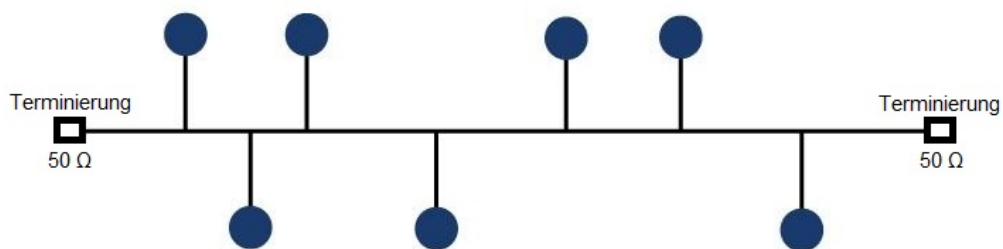


Abbildung 2.11.: Ethernet-Bus-Topologie mit RG58 Koaxialkabel als Übertragungsmedium.

### 2.2.2.2. Stern-Topologie

Im Unterschied zur Bus-Topologie sind die Geräte in einer Stern-Topologie an einem zentralen Punkt (Hub oder Switch) angeschlossen, welcher als Verteilstation dient.

Vorteile der Stern-Topologie sind hohe Übertragungsraten (beim Switch), Skalierbarkeit und Stabilität. Nachteile sind der hohe Aufwand für die Verkabelung und die Tatsache, dass es beim Ausfall des zentralen Gerätes zum gleichzeitigen Ausfall von allen Verbindungen zwischen den Teilnehmern des Netzwerkes kommt. Das kann durch Verdoppeln des zentralen Gerätes (Redundanz) verhindert werden.

## 2. Grundlagen

Beispiele für Netzwerke mit Stern-Topologie sind Fast Ethernet, Fibre-Channel, Infini-Band und Token Ring[[1]]. Die Abb. 2.12 zeigt eine Ethernet-Stern-Topologie.

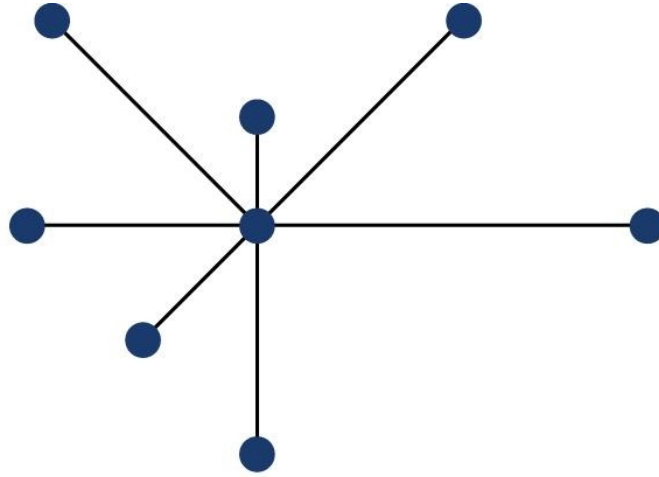


Abbildung 2.12.: Ethernet-Stern-Topologie



## 3. Ethernet im industriellen Bereich

*In diesem Kapitel wird das Thema Industrial Ethernet sowie die verschiedenen Ethernet-Lösungskonzepte unter die Lupe genommen.*

### 3.1. Industrial Ethernet

Industrial Ethernet wurde eingeführt, um eine einheitliche Kommunikationsinfrastruktur von der Betriebsführungsebene bis zum Aktor/Sensor-Bus zu schaffen.

Die Anzahl der bislang entwickelten Ethernet-Lösungen übertrifft zwischenzeitlich die Anzahl der Datenbussysteme. Entscheidende Unterschiede zwischen den jeweiligen Industrial-Ethernet-Systemen liegen in der Organisation der Datenübertragung und der Erzeugung des Echtzeitverhaltens[[8]].

#### 3.1.1. Anforderungen

Auf Basis von Ethernet lassen sich durchgängige Kommunikationsinfrastrukturen realisieren. Diese ermöglichen einen direkten Datenaustausch zwischen allen Schichten der Automatisierungspyramide und der IT-Welt in einem Unternehmen (Vertikale Integration).

Dafür müssen Ethernet-Lösungen zwei Voraussetzungen erfüllen: Sie müssen standardisierte Kommunikationsprotokolle und -dienste der IT-Welt unterstützen und die Echtzeitanforderungen der Automation erfüllen[[8]].

Anforderungen an die Echtzeitfähigkeit hängen vom jeweiligen Einsatzgebiet und Anwendungsfall ab. Üblicherweise werden heute drei Anforderungsklassen unterschieden, die durch Aktualisierungsraten der Prozessdaten definiert werden können:

- Applikationen mit keinen oder geringen Anforderungen an die Echtzeit. Aktualisierungsraten von ungefähr 100 ms.
- Applikationen mit hohen Anforderungen an die Echtzeit und typischen Aktualisierungsraten von bis zu 10 ms. In dieser Kategorie befinden sich Anwendungen ethernet-basierter Kommunikationssysteme in der Automatisierungstechnik (z.B. Fertigungsautomation).
- Applikationen mit harten Anforderungen an die Echtzeit. Typische Aktualisierungsraten von weniger als 1 ms. Ein Beispiel sind Motion-Control-Anwendungen[[8]].

### 3. Ethernet im industriellen Bereich

#### 3.1.2. Lösungskonzepte

In Abhängigkeit von den Echtzeit- und Kostenanforderungen, lassen sich die verschiedenen Real-Time-Ethernet Konzepte in drei Kategorien (Abb. 3.1) einteilen:

- **Klasse A:** Verwendet Standard, unmodifizierte Ethernet-Hardware sowie Standard-TCP/IP-Stacks für die Datenkommunikation. Die Echtzeitfähigkeit ist durch die unvorhersehbaren Verzögerungen bei der Infrastruktur (z.B. Switches) begrenzt. Das größte Hindernis für eine bessere Echtzeit-Leistung ist jedoch der TCP/IP-Stack. In dieser Klasse befinden sich PROFINET CbA, EtherNet/IP und Modbus-TCP[[5]].
- **Klasse B:** Lösungskonzepte der Klasse B verwenden noch immer Standard Ethernet-Hardware, aber verzichten auf die Verwendung von TCP/IP für die Prozessdatenkommunikation. Das Protokoll setzt direkt auf dem Ethernet-Frame auf und die Adressierung der Teilnehmer erfolgt über die MAC-Adressen der Netzwerkkarten. Durch den geringeren Protokoll-Overhead kommt es zu einem verbesserten Zeitverhalten mit höherer Deterministik und geringerem Jitter. Zu dieser Kategorie zählen POWERLINK und PROFINET RT.
- **Klasse C:** Im Gegensatz zur Klasse B wird zusätzlich das Ethernet-Buszugriffsverfahren (Timingverhalten) modifiziert und spezifische Ethernet-Hardware eingesetzt. Die Lösungskonzepte dieser Klasse erfüllen die höchsten Anforderungen an das Zeitverhalten von Echtzeit-Ethernet (Aktualisierungsintervall  $<1$  ms, Jitter  $<1$   $\mu$ s). In dieser Klasse befinden sich PROFINET IRT, EtherCAT, POWERLINK, Sercos und CC-Link IE[[8]].

In weiterer Folge werden fünf (POWERLINK, EtherNet/IP, EtherCAT, PROFINET und Modbus-TCP) von weltweit ungefähr 30 verwendeten Industrial-Ethernet-Systemen näher beschrieben.

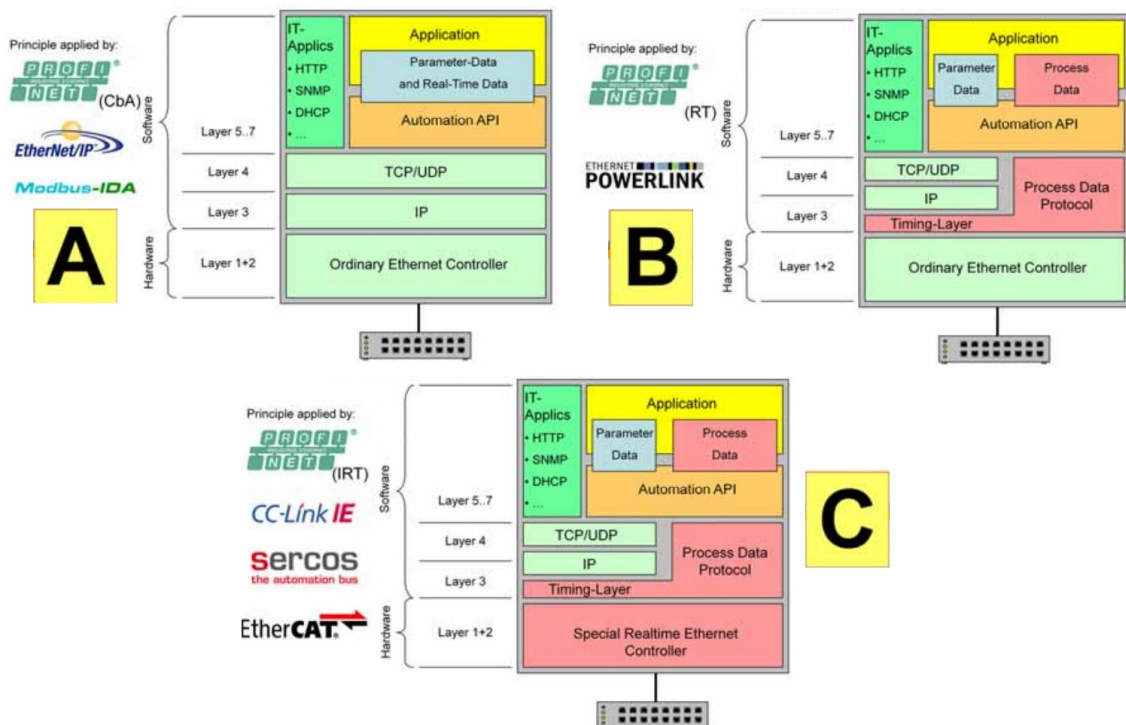


Abbildung 3.1.: Die Echtzeit-Ethernet-Konzepte werden in drei Kategorien unterteilt[[5]]

### 3.1.2.1. PROFINET

PROFINET<sup>1</sup> (Process Field Network) ist ein offener Industrial Ethernet-Standard, welcher von der Firma Siemens und den Mitgliedsfirmen der PROFIBUS-Nutzerorganisation (PNO) entwickelt wurde.

PROFINET ist der Nachfolger des Industriestandards PROFIBUS. Das PROFINET Konzept basiert auf Ethernet-TCP/IP, ist echtzeitfähig und ermöglicht die Integration von Feldbussystemen. PROFINET ergänzt PROFIBUS für Anwendungen, bei denen schnelle Datenkommunikation über Ethernet in Kombination mit industriellen IT-Funktionen gefordert wird[[8]].

<sup>1</sup><http://w3.siemens.com/mcms/automation/de/industrielle-kommunikation/profinet/seiten/default.aspx/>, Zugriffsdatum: 11.03.2018

### 3. Ethernet im industriellen Bereich

Das PROFINET-Konzept ist modular aufgebaut (Abb. 3.2) und die Varianten unterscheiden sich im Wesentlichen durch die Art des Datenaustausches:

- **PROFINET CBA** (Component Based Automation): Wurde für die komponentenbasierte Kommunikation über Ethernet-TCP/IP zwischen intelligenten Automatisierungsgeräten konzipiert. Der Ansatz von PROFINET CBA ist die verteilte Automatisierung, wobei eine Aufgabe auf mehrere Geräte verteilt wird.  
Diese Geräte sind über Ethernet miteinander verbunden und lösen gemeinsam die Automatisierungsaufgabe. PROFINET CBA wird auch als Non-Realtime bezeichnet und ist für Echtzeitanwendungen eher ungeeignet (Buszykluszeiten von 50-100 ms).
- **PROFINET IO**: Die PROFINET IO (Input-Output) Technologie ist für die Kommunikation zwischen einer Steuerung und den dezentralen Bussystemen konzipiert. PROFINET IO baut auf dem Funktionsmodell von Profibus-DP auf. Das Konzept ist für die schnelle Übertragung von I/O-Daten geeignet und bietet zeitgleich eine Übertragungsmöglichkeit für Parameter und Bedarfsdaten sowie IT-Funktionen. PROFINET IO bietet zusätzliche Funktionen für die Übertragung von Parametrierungen, Diagnosen und Alarmen.  
Die PROFINET RT-Kommunikation für keine oder weiche Echtzeit stellt die Basis für den Datenaustausch bei PROFINET IO dar. Dabei werden Echtzeit-Daten gegenüber TCP-UDP/IP-Daten höher-prioritär behandelt. Damit werden Buszykluszeiten im Bereich von wenigen Millisekunden möglich (<10 ms). PROFINET IRT (Isochronous Real Time) für harte Echtzeit (Echtzeitkategorie 3) definiert den taktsynchronen Datenaustausch mit PROFINET und wird z.B. bei Motion-Control-Anwendungen eingesetzt.  
Im Unterschied zur Echtzeit-Kommunikation wird der Beginn eines Buszyklus mit höchster Präzision eingehalten. Dabei kann der Beginn eines Buszyklus maximal um 1  $\mu$ s abweichen. Buszykluszeiten liegen im Bereich von wenigen hundert Mikrosekunden bis zu einer Millisekunde[[8]].

Mit PROFINET können Lösungskonzepte für die Prozessautomatisierung, Fertigungstechnik, Gebäudeautomation sowie für das gesamte Spektrum der Antriebstechnik realisiert werden. PROFINET ist in der IEC 61158 und IEC 61784 standardisiert. Es steht das Sicherheitsprofil PROFIsafe zur Verfügung.

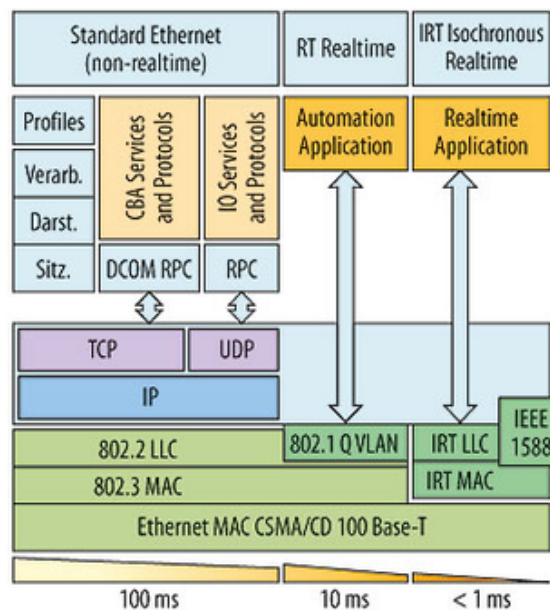


Abbildung 3.2.: PROFINET nutzt eine dreistufige Echtzeitskalierung[[18]]

### 3.1.2.2. EtherCAT

EtherCAT<sup>2</sup> (Ethernet for Control and Automation Technology) ist ein ethernet-basiertes Bussystem, welches von der Firma Beckhoff Automation und der ETG (EtherCAT Technology Group) entwickelt wurde.

EtherCAT ist eine offene Technologie die in IEC 61158 und IEC 61784 sowie in ISO 15745-4 standardisiert ist[[8]].

EtherCAT basiert auf dem Summenrahmenverfahren: Der Master sendet einen Ethernet-Frame mit den Daten für alle verbundenen Slaves, der nacheinander alle Teilnehmer durchläuft. Beim letzten Teilnehmer eines Strangs wird der Frame wieder zurückgeleitet. Der Frame wird während des Durchlaufs von den Teilnehmern in einer Richtung bearbeitet. Die für den Teilnehmer bestimmten Daten werden *on the fly* ausgelesen und in den Frame eingefügt[[6]].

Dieses Summenrahmenprotokoll vermeidet den üblichen großen Overhead bei kleinen Nutzdatenmengen und schafft ein sehr effizientes System, das gerade kleinere Datenmengen extrem schnell übertragen kann[[18]].

Im Gegensatz zu Protokollen wie POWERLINK oder PROFINET, beschreibt EtherCAT ausschließlich die Layer 1–3 des OSI-7-Schichten-Modells.

<sup>2</sup><https://www.ethercat.org/>, Zugriffsdatum: 11.03.2018

### 3. Ethernet im industriellen Bereich

Für den Echtzeit-Datenaustausch (Abb. 3.4) wird über den EtherCAT Slave Controller eine CANopen-Kommunikation initiiert. Über CoE (CAN over EtherCAT) kann eine PDO/SDO-Kommunikation durchgeführt werden, wobei auch die CANopen-Profile unterstützt werden. Für den asynchronen Datenaustausch steht ein Mailbox-System zur Verfügung, über das andere Protokolle wie z.B. EoE (Ethernet over EtherCAT) oder FoE (File Access over EtherCAT) getunnelt werden können[[18]].

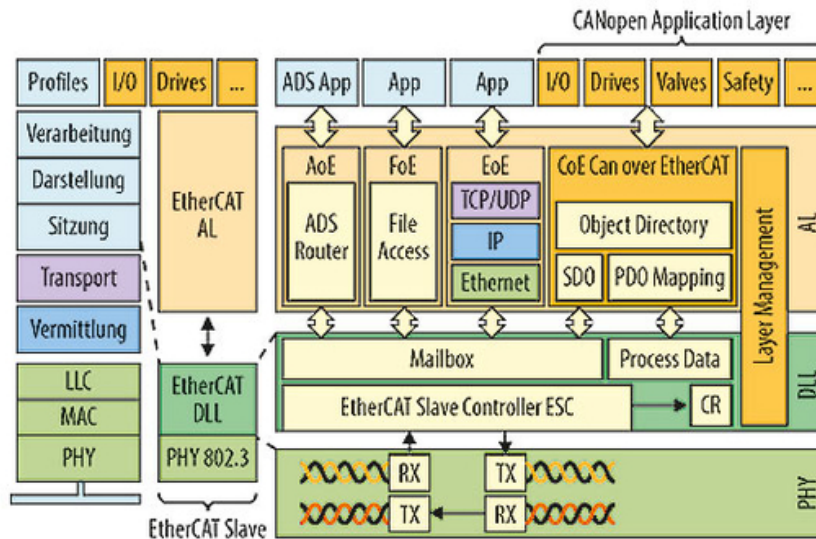


Abbildung 3.3.: EtherCAT-Slaves haben einen sehr schlanken Protokoll-Stack, der beliebige andere Protokolle tunneln kann[[18]]

EtherCAT eignet sich insbesondere für schnelle Anwendungen wie z.B. Verpackungsmaschinen, Spritzgussmaschinen, schnelle Pressen, CNC-Bearbeitungszentren, Robotik und Hydraulikregelungen. Die EtherCAT Mechanismen und der Verzicht auf eine Laufzeit-beeinflussende Infrastruktur sowie die effiziente Übertragung auch kleiner Datenmengen ermöglichen einen Datenaustausch, bei dem nicht mehr das Bussystem, sondern der Master der begrenzende Faktor ist[[18]].

#### 3.1.2.3. Modbus-TCP

Modbus-TCP<sup>3</sup> ist ein weit verbreitetes industrielles Kommunikationsprotokoll, das 1979 von Gould-Modicon entwickelt wurde.

Modbus-TCP ist ein Client/Server-Protokoll für den verbindungsorientierten Austausch von Prozessdaten. Es ist einfach zu implementieren, effektiv und für Anbieter als auch für Anwender frei verfügbar[[8]].

<sup>3</sup><http://www.modbus.org/>, Zugriffsdatum: 11.03.2018

Das Modbus-Protokoll ist in der Anwendungsschicht (Schicht 7) des ISO/OSI-Modell implementiert. Es ermöglicht eine Client/Server-Kommunikation zwischen verschiedenen Geräten die über unterschiedliche Bussysteme oder Netzwerke verbunden sind (Abb. 3.4).

Das Protokoll ist unabhängig von den darunter liegenden physikalischen Schichten. Daher existieren unterschiedliche Implementierungen des Protokolls auf verschiedenen Kommunikationsmedien (z.B. Ethernet, RS-232, Funkstrecken, Glasfasern)[[8]].

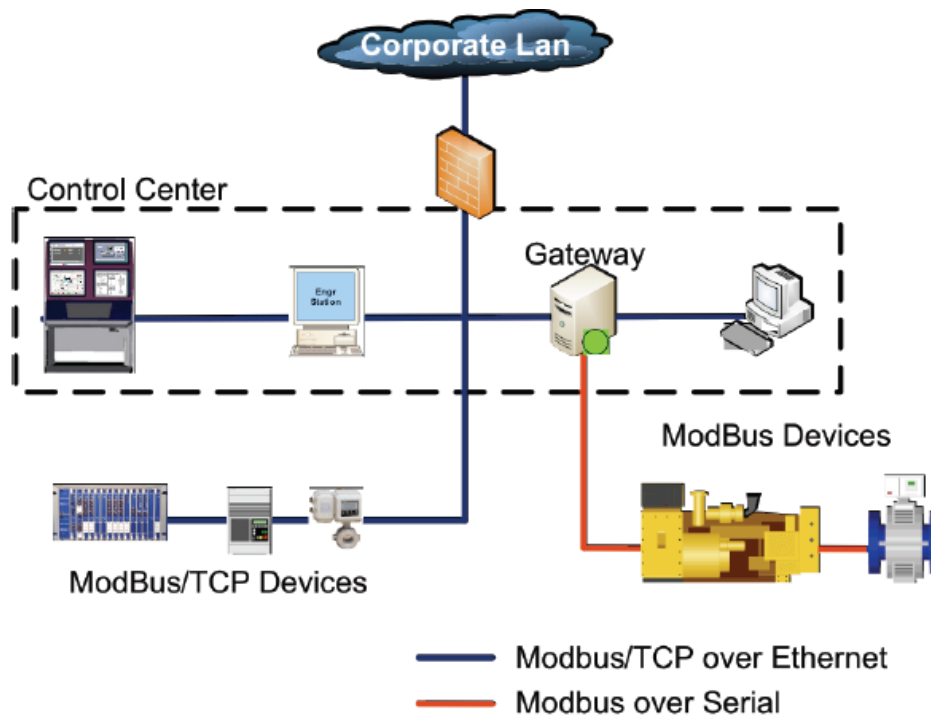


Abbildung 3.4.: Modbus-TCP Architektur[[6]]

Das Modbus-Protokoll kann auf allen Gerätetypen eingesetzt werden wie z.B. auf Speicherprogrammierbaren Steuerungen, Motion-Controllern, Bedienterminals, in SCADA-Systemen sowie auf allen Peripherie-Geräten die im Automatisierungs-Bereich genutzt werden[[8]].

#### 3.1.2.4. Ethernet POWERLINK

Das Industrial-Echtzeit-Protokoll POWERLINK<sup>4</sup> wurde von der Firma B&R entwickelt und wird heute von EPSG (Ethernet Powerlink Standardization Group) als offener Standard weiterentwickelt.

<sup>4</sup><http://www.ethernet-powerlink.org/>, Zugriffsdatum: 11.03.2018



### 3. Ethernet im industriellen Bereich

POWERLINK wird für Anwendungen im Maschinen- und Anlagenbau sowie für Anwendungen in der Prozessindustrie eingesetzt.

Die POWERLINK-Implementierung beinhaltet eine IEEE 802.3-konforme Adressierung der Slaves, eine Integration der TCP/IP-Kommunikation für den Austausch asynchroner Daten und die Nutzung von CANopen als Anwendungs-Layer (siehe Abb. 3.5). Damit bietet POWERLINK ein alternatives High-Speed-Kommunikationsmedium zu der konventionellen CAN-Anschaltung[[18]].

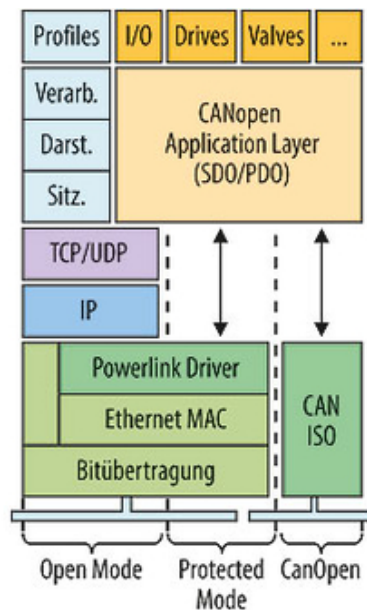


Abbildung 3.5.: Beim POWERLINK-Protokollstapel ist CANopen das Bindeglied zu anderen Systemen[[18]]

Ein Powerlink-Netzwerk wird durch einen Master (Managing Node) arbitriert. Ein Kommunikationszyklus besteht aus einem isochronen- und asynchronen-Abschnitt. In der isochronen Phase erfolgt eine exakt determinierte Echtzeitkommunikation. In der asynchronen kann eine konventionelle IP-Kommunikation mit den Feldgeräten durchgeführt werden.

2012 führte die EPSG das *Poll Response Chaining* ein, bei dem der Master alle seine Daten über *Poll Response Broadcast* auf den Bus legt und die Slaves in der festgelegten Reihenfolge antworten. Damit kann eine signifikante Geschwindigkeitszunahme bei vielen Teilnehmern erreicht werden[[18]].

POWERLINK kann sowohl als Softwarelösung ohne Hardwarebeschleunigung für Anforderungen mit weicher Echtzeit als auch als High-End-System mit Prozessor-Unterstützung für Zykluszeiten bis zu 100  $\mu$ s eingesetzt werden[[8]].



## 3.1.2.5. EtherNet/IP

EtherNet/IP<sup>5</sup> (Ethernet Industrial Protocol) ist ein offener industrieller Standard der von Allen Bradley (Rockwell Automation) und der ODVA (Open DeviceNet Vendor Association) entwickelt und in der internationalen Normenreihe IEC 61158 standardisiert wurde.

EtherNet/IP nutzt die Technologie von Ethernet-TCP/IP und als Übertragungsmedium kommen Ethernet-Twisted-Pair-Kabel oder Lichtwellenleiter zum Einsatz. In einem EtherNet/IP-Netzwerk werden die Geräte über eine Punkt-zu-Punkt-Verbindung an einen Switch angeschlossen[[8]].

Bei EtherNet/IP wurde das Anwendungsprotokoll CIP (Common Industrial Protocol) (Abb. 3.6) auf das Transportprotokoll Ethernet portiert. CIP wird auch bei DeviceNet und ControlNet eingesetzt.

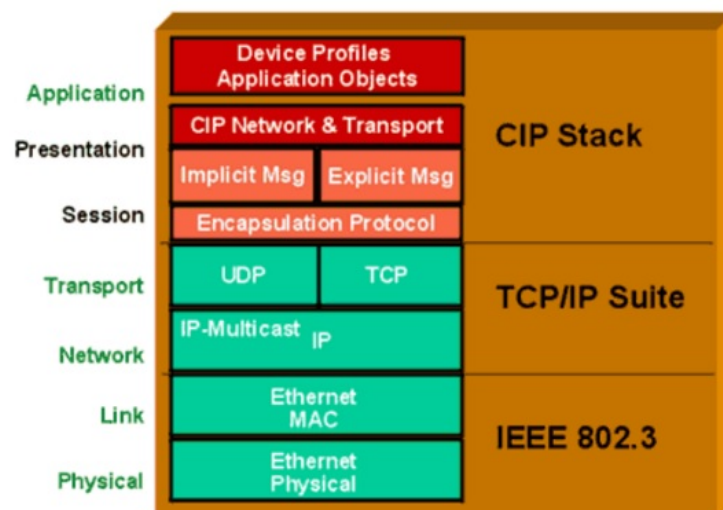


Abbildung 3.6.: Schichtenmodell des EtherNet/IP-Protokollstapel[[8]]

EtherNet/IP verwendet Standard-Ethernet-Hardware und nutzt zur Datenübertragung TCP-UDP/IP. Bei EtherNet/IP stehen verschiedene Kommunikationsmechanismen zur Verfügung, wie z.B. zyklisches Polling, Multicast und zeit- oder ereignisgesteuerte Auslösung. CIP unterscheidet zwischen impliziten E/A-Nachrichten und expliziten Frage/Antwort-Telegrammen zur Konfiguration und Datenerfassung. Explizite Nachrichten werden in TCP-Frames eingebettet, während Daten für Echtzeitanwendungen wegen des kleineren Overheads per UDP versendet werden[[6]].

<sup>5</sup><https://www.odva.org/Technology-Standards/EtherNet-IP/Overview/>, Zugriffsdatum: 11.03.2018

### 3. Ethernet im industriellen Bereich

Neben der Übertragung von Prozessdaten und Parameter unterstützt EtherNet/IP auch die industriellen IT-Funktionen wie Web-Server und E-Mail-Client. Es steht das Sicherheitsprofil CIP Safety zur Verfügung.

#### 3.1.3. Datenbussysteme im Vergleich

Die technischen Unterschiede zwischen den Datenbussystemen bestehen in der maximalen Anzahl von Datenbytes je Telegramm, der erreichbaren Kabellänge, sowie beim Funktionsumfang. Die Grundfunktionen aller Bussysteme sind die schnelle zyklische Übertragung von Eingangs- und Ausgangsdaten.

Datenbusse werden heute in allen Branchen verwendet, von der Prozess- und Fertigungsautomatisierung, über den Maschinenbau bis hin zum Einsatz in explosionsgefährdeten Bereichen. Die immer leistungsfähigeren ethernet-basierten Systeme werden die Bussysteme in naher Zukunft ersetzen.

Die Tabelle 3.7 gibt eine Übersicht über die gängigen industriellen Bussysteme.

	AS-Interface	CANopen	CC-Link	ControlNet	DeviceNet	Interbus	Profibus
<b>Anwendungsbereiche</b>	Sensor- / Aktorebene	Feldebene	Feldebene	Zellenebene	Feldebene	Feldebene	Feldebene
<b>Besonderheiten</b>	schnell, kostengünstig, ideal für rein binäre E/A-Signale	ideal für kleine Datenmengen und schnelle Synchronisation	hohe Geschwindigkeit, deterministisch, hohe EMV-Immunität	hohe Übertragungsleistung bei gleichbleibenden Buszykluszeiten	basiert auf CAN, überträgt zusätzlich 24 Volt im Buskabel	sehr schnelle, gute Diagnose	mehrere Varianten mit abgestufter Funktionalität: DP, DPV1, DPV2, PA
<b>Reaktionszeiten</b>	klein	mittel	mittel	klein	mittel	klein	mittel
<b>Topologie</b>	Linie, Stern, Baum	Linie	Linie, Stern, T-Branch	Linie	Linie	Ring	Linie
<b>Übertragungsmedium</b>	Kabel (2-adrig, ungeschirmt)	Kabel (4-adrig, geschirmt, verdrillt)	Kabel (3-adrig, geschirmt, verdrillt), Kabel (5-adrig, geschirmt, verdrillt, inkl. Spannungsversorgung)	Koaxialkabel	Kabel (4-adrig, geschirmt, verdrillt)	Kabel (5-adrig, geschirmt, verdrillt), Lichtwellenleiter	Kabel (2-adrig, geschirmt, verdrillt), Lichtwellenleiter
<b>Übertragungsverfahren (Physical Layer)</b>	alterierende Pulsmodulation	CAN	RS-485	Modulation	CAN	RS-485, LWL	RS-485, LWL oder IEC 61158-2 (MBP)
<b>Übliche Anschlusstechniken</b>	Schneidklemmtechnik	Sub-D (9-polig)	Klemme, Sub-D (9-polig)	BNC	Klemme, Mini- und Micro-Style	Sub-D (9-polig)	Sub-D (9-polig), M12 für IP65
<b>Max. Teilnehmerzahl</b>	62	127	64	99	64	256	126
<b>Max. Ausdehnung je Segment</b>	100 m	100 m (500 kbit/s), 1000 m (62,5 kbit/s)	100 m (10 Mbit/s), 1200 m (156 kbit/s)	1000 m	100 m (500 kbit/s), 1000 m (62,5 kbit/s)	zwischen 2 Geräten max. 400 m	100 m (12 Mbit/s), 1200 m (9,6 kbit/s)
<b>Energieversorgung der Teilnehmer über den Bus</b>	ja, 30 Volt	optional 24 Volt	optional 24 Volt	nein	ja, 24 Volt	nein	DP: nein PA: optional
<b>Einsatz im Exbereich</b>	bedingt	nein	nein	bedingt	nein	nein	DP: nein PA: ja

Abbildung 3.7.: Unterschiede zwischen industriellen Bussystemen [[8]]

### 3. Ethernet im industriellen Bereich

#### 3.1.4. Safety Aspekte

Sicherheitsgerichtete Automatisierungslösungen müssen heute neben Safety-Aspekten (funktionale Sicherheit) auch Security-Aspekte (Cyber Security) berücksichtigen. Die funktionale Sicherheit hat dabei die Aufgabe den Mensch, die Maschine und die Umwelt zu schützen. Die Cyber Security hingegen hat zur Aufgabe die Verfügbarkeit, Integrität und Vertraulichkeit von Daten der Anlage sicherzustellen. Erst das Zusammenspiel zwischen der funktionaler Sicherheit und der Informationssicherheit gewährleistet die Gesamtsicherheit der Anlage[[8]].

Wie schon bei Industrial Ethernet haben sich auch im Safety-Bereich verschiedene Standards etabliert (Abb. 3.8). Für PROFINET steht das Sicherheitsprofil PROFIsafe zur Verfügung, welches auch eine drahtlose sicherheitsgerichtete Kommunikation ermöglicht. POWERLINK und openSAFETY stellen kostenlos die *Sourcecodes* für Master und Slave bereit. CIP Safety wird von EtherNet/IP und Sercos genutzt. EtherCAT setzt auf das offene Protokoll Safety-over-EtherCAT (FSoE (FailSafe over EtherCAT)).



Abbildung 3.8.: Industrielle Ethernet-Protokolle zur Übertragung von sicherheitsrelevanten Daten[[8]]

Das Sicherheitsprotokoll hat die Aufgabe die Integrität der Datenübertragung zu gewährleisten und ständig den Übertragungsprozess zu überwachen. Dabei sollen folgende Fehler vermieden werden: Dateneinfügungen, Datenverluste, Datenverfälschungen und Datenwiederholungen. Ebenso müssen die falsche Abfolge von Daten oder übermäßige Verzögerungen rechtzeitig erkannt werden. Aus diesem Grund werden die sicherheitsrelevanten Segmente des Netzwerks und die Funktion der dazugehörigen Geräte vom Safety-System zyklisch auf fehlerfreie Funktion überprüft. Bei einer unvollständigen Datenübertragung oder einer Unterbrechung der Kommunikation werden vom Safety-System die entsprechenden Sicherheitsfunktionen oder die sichere Abschaltung der Anlage eingeleitet[[6]].

#### 3.1.5. Embedded TCP/IP-Stacks

Aufgrund von Einschränkungen durch die Hardware der Embedded-Geräte, wie geringer Speicher und niedrige oder marginale Rechenleistung, verwenden diese spezielle TCP/IP-Stacks. Die Mehrheit der verfügbaren TCP/IP-Stacks kann mit oder ohne ein Betriebssystem betrieben werden. Auf dem Markt sind zahlreiche Open-Source- und kommerzielle Implementierungen erhältlich.

Anforderungen an einen TCP/IP-Protokollstack für Embedded-Systeme:

- Der TCP/IP-Stack soll nicht unnötig viel Speicher und Prozessor-Rechenzeit des Embedded-Systems verbrauchen.
- *Zero-Copy-Stack*: Bedeutet, dass neue Daten nur einmal in einen Puffer kopiert werden und die einzelnen Protokollebenen direkt auf diesem Puffer arbeiten.
- Der TCP/IP-Stack sollte modular sein (Abb. 3.9), da nicht alle Anwendungen die volle Funktionalität einer kompletten TCP/IP-Implementierung benötigen.
- Da in Embedded-Systemen die verschiedensten Prozessoren, Softwarekonzepte und Betriebssysteme zum Einsatz kommen, sollte ein Embedded TCP/IP-Stack für alle Arten von Betriebssystemen kompatibel sein.
- Der TCP/IP-Stack sollte nur einen *Timer* mit einer sehr flexibel einstellbaren Time-rauflösung benötigen, um eine einfache Integration in ein bestehendes System zu ermöglichen.
- Die Puffer sollen nicht während der Laufzeit dynamisch alloziert werden, stattdessen soll mit sogenannten *preallocated* Puffern gearbeitet werden.
- Der TCP/IP-Stack sollte mit unterschiedlichen Arten von physikalischen Schnittstellen (Ethernet, serielle Schnittstelle) zusammenarbeiten können[[33]].

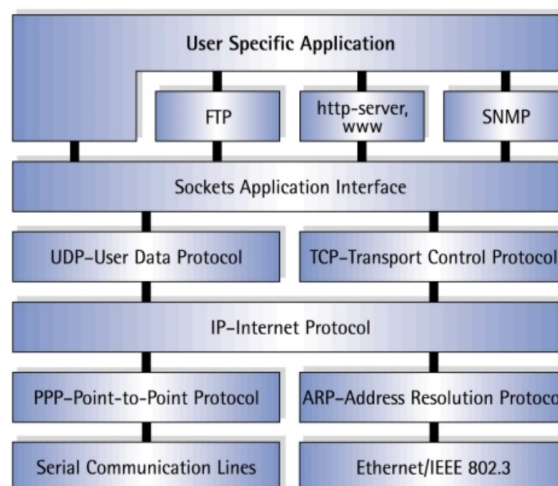


Abbildung 3.9.: Struktur eines Modularen TCP/IP-Stacks für Embedded-Systeme[[33]]

### 3. Ethernet im industriellen Bereich

#### 3.1.5.1. Open-Source-Implementierungen

##### **lwIP**

Das lightweight Internet Protocol (lwIP)<sup>6</sup> ist eine unabhängige Implementierung des TCP/IP-Stacks, optimiert für den Embedded-Bereich. Der Fokus liegt auf der Reduzierung der benötigten Ressourcen, während der TCP/IP noch immer im vollen Ausmaß verfügbar bleibt. lwIP nutzt ungefähr 40 kB RAM (Random Access Memory)- und 30 kB ROM (Read Only Memory)-Speicher. Der lwIP TCP/IP-Stack unterstützt folgende Protokolle: IPv4, IPv6, UDP, TCP, ICMP, IGMP, SNMP, ARP und PPP. lwIP ist weit verbreitet und wird oft eingesetzt.

##### **uIP**

uIP (micro IP)<sup>7</sup> ist die kleinste Open-Source-Implementierung des TCP/IP-Stacks für die Verwendung mit 8- und 16-Bit-Mikrocontrollern. uIP ist für kleine Mikrocontroller-Systeme vorgesehen, bei denen die Code-Größe und der RAM-Speicher stark eingeschränkt sind (erfordert nur 4-5 kB ROM und 1 kB RAM). uIP ist jetzt Teil von Contiki, einem sehr kleinen, portablen Betriebssystem für Embedded-Geräte.

##### **FNET**

FNET<sup>8</sup> ist ein Open-Source, Dual-TCP/IPv4 und IPv6-Stack (lizenziert unter Apache Version 2.0) der Firma Freescale für Embedded-Systeme, welche auf der 32 Bit-Architektur basieren. Es ist eine komplett eigenständige Implementierung, d.h. der funktionstüchtige Gebrauch ist ohne ein Betriebssystem möglich. FNET unterstützt die wesentlichen Teile der Internetprotokollfamilie: UDP, TCP, IP, ICMP und ARP.

##### **CycloneTCP Open**

CycloneTCP Open<sup>9</sup> ist ein Dual-IPv4/IPv6-Stack optimiert für Embedded-Anwendungen. Durch die Unterstützung von IPv6, erleichtert CycloneTCP den Übergang zum *Next Generation Internet*. CycloneTCP unterstützt Industriestandard-Mikrocontroller mit integriertem MAC als auch *standalone* Ethernet-Controller. Der TCP/IP-Stack ist entweder als Open-Source (CycloneTCP Open) oder unter einer kommerziellen Lizenz (CycloneTCP Lite, Pro oder Ultimate) erhältlich.

---

<sup>6</sup><http://savannah.nongnu.org/projects/lwip/>, Zugriffsdatum: 11.03.2018

<sup>7</sup><https://sourceforge.net/projects/uip-stack/>, Zugriffsdatum: 11.03.2018

<sup>8</sup><http://fnet.sourceforge.net/>, Zugriffsdatum: 11.03.2018

<sup>9</sup>[http://www.oryx-embedded.com/cyclone\\_tcp.html/](http://www.oryx-embedded.com/cyclone_tcp.html/), Zugriffsdatum: 11.03.2018

### 3.1.5.2. Kommerzielle Implementierungen

Im Gegensatz zu Open-Source-Implementierungen, versprechen alle kommerziellen TCP/IP-Stacks eine hohe Robustheit und Effizienz. Weiters nutzen alle kommerziellen Implementierungen den *Zero-Copy-Mechanismus*, um eine effiziente Nutzung der Ressourcen zu gewährleisten.

#### uC/TCP-IP

uC/TCP-IP-Stack<sup>10</sup> der Firma Micrium ist ein kompakter *High-Performance* TCP/IP-Stack, welcher für Embedded-Systeme optimiert ist, vorwiegend für den Einsatz in sicherheitskritischen Anwendungen. uC/TCP-IP läuft auf uC/OS-II und verwendet etwa 100 kB ROM und 60 kB RAM Speicher. Durch die Verwendung von großen und kleinen Puffer, wird die Leistung des Systems maximiert.

#### NicheStack

Die TCP/IP-Protokollfamilie NicheStack<sup>11</sup> der Firma INTERNICHE ist für kleine (8/16 Bit) als auch für große (32 Bit) Embedded-Systeme geeignet. Der NicheStack enthält die komplette Funktionalität, die in den RFCs spezifiziert ist und bietet auch die Möglichkeit, nicht benötigte Eigenschaften abzuschalten. Die Software ist mit oder ohne Betriebssystem einsetzbar und erfordert etwa 50 kB ROM- und RAM-Speicher. Ergänzt wird der TCP/IP-Stack durch eine Vielfalt von spezifischen Anwendungen, wie Web-Server, FTP-Server, E-Mail Alerter usw. NicheLite ist ein neuer TCP/IP-Stack, welcher einen vollständigen prozessorunabhängigen TCP/IP-Protokollstack mit nur 12 kB realisiert[[33]].

#### smxNS

smxNS<sup>12</sup> ist ein robuster und kompakter TCP/IP-Stack des Unternehmen Micro Digital für Embedded Systeme. Er ist in C geschrieben und kann auf jeder Hardwareplattform eingesetzt werden. smxNS kann so konfiguriert werden, dass nur die Teile (Clients, Server, Protokolle, Treiber), die tatsächlich in der Anwendung benötigt werden, einbezogen werden. Somit kann der smxNS TCP/IP-Stack auf ungefähr 25 kB Code-Größe reduziert werden.

#### CMX-TCP/IP

CMX-TCP/IP<sup>13</sup> ist eine portable TCP/IP-Implementierung für Embedded-Systeme, welche mit oder ohne einen CMX-RTX RTOS (Real-Time Operating System) laufen kann. CMX-TCP/IP unterstützt eine große Anzahl an Prozessoren und verbraucht ungefähr 20 kB ROM Speicher. Es sind Testanwendungen verfügbar wie z.B. ein Benchmark-Server oder ein Ping-Tester.

<sup>10</sup><https://www.micrium.com/rtos/uctcpip/overview/>, Zugriffsdatum:11.03.2018

<sup>11</sup><http://www.iniche.com/nichestack.php/>, Zugriffsdatum:11.03.2018

<sup>12</sup><http://www.embedded-tools.de/smxNS/>, Zugriffsdatum:11.03.2018

<sup>13</sup><http://www.cmx.com/tcpip.htm/>, Zugriffsdatum:11.03.2018



### 3. Ethernet im industriellen Bereich

#### 3.2. Industrial Wireless

Überall dort, wo eine kabelgebundene Verbindung (Feldbusse, Industrial Ethernet) nicht möglich oder ineffizient ist (z.B. bewegliche Systeme, unzugängliche Umgebung), kommen heute verschiedene Funklösungen zum Einsatz. In der Industrie und industrienahen Bereichen bietet die Drahtlose Kommunikation zahlreiche Vorteile:

- Hohe Flexibilität, sowie einfache und zuverlässige Kommunikation mit beweglichen oder schwer zugänglichen Anlagen.
- Einfache Installation und geringer Material- und Kostenaufwand gegenüber kabelgebundenen Lösungen.
- Reduzierung von Wartungskosten (z.B. verschleißanfällige Kabelstränge) und einfache Erweiterung des bestehenden Netzwerkes.
- Steuerung und Überwachung von entfernten Stationen über Funk[[24]].

Drahtlose Netzwerke weisen jedoch Nachteile gegenüber kabelgebundenen Applikationen hinsichtlich Zuverlässigkeit, Echtzeiteigenschaften und Datensicherheit auf.

In der industriellen Umgebung gibt es verschiedene Anwendungsbeispiele (Abb. 3.10) für Funktechnologien: von der Übertragung eines einfachen Sensorwerts bis hin zu einem Netzwerk mit hunderten von Teilnehmern. Einige Anwendungsbeispiele aus der Fertigungsautomatisierung sind Elektrohängebahnen, Fahrerlose Transportsysteme und Lager-Shuttles, Portalkrane, Regalbediengeräte usw. [[24]].



Abbildung 3.10.: Anwendungsbereiche von Funksystemen in der industriellen Umgebung [[24]]



### 3.2.1. Wireless-Technologien

Durch die verschiedenen Anwendungsbereiche in der Industrie, ergeben sich unterschiedliche Anforderungen an die Funktechnologien. Diese Anforderungen können zur Zeit von keiner Funktechnologie umfassend abgedeckt werden. Die wichtigste Voraussetzung für den Einsatz von Funktechnologien ist, dass sie unter schwierigen Bedingungen ebenso robust und zuverlässig arbeiten, wie eine kabelgebundene Verbindung.

Für die drahtlose Übertragung kommen WLAN, Bluetooth, Zigbee, Mobilfunk und WirelessHART in Betracht. Aufbauend auf diesen Funktechnologien werden die industriellen Ethernet-Protokolle (z.B. PROFINET, EtherNet/IP) aufgesetzt[[24]]. Ein bestehendes Ethernet-Netzwerk kann mit wenig Aufwand um ein drahtloses Netz erweitert werden (siehe Abb. 3.11).



Abbildung 3.11.: Drahtlose Erweiterungsmöglichkeiten eines Industriellen Ethernet-Netzwerkes[[24]]

Wireless Technologien:

- **WLAN:** Ist nach IEEE 802.11 a/b/g/n<sup>14</sup> standardisiert. WLAN wird z.B. für mobile Anwendungen wie fahrerlose Transportsysteme oder für die Datenerfassung im Lagerbereich eingesetzt. WLAN ermöglicht Datenraten bis zu 300 Mbit/s und eine hohe Zuverlässigkeit durch die *Multiple Input Multiple Output* (MiMO)-Technik.
- **Bluetooth:** Kommt dann zum Einsatz, wenn wenige Teilnehmer mit niedrigen Übertragungsraten über kurze Distanzen hinweg miteinander kommunizieren sollen. Eigenschaften von Bluetooth sind eine extrem zuverlässige Übertragung, Reichweiten bis zu 200 m und geringe Verzögerungszeiten. Bluetooth ist nach IEEE 802.15.1<sup>15</sup>

<sup>14</sup><http://www.ieee802.org/11/>, Zugriffsdatum: 11.03.2018

<sup>15</sup><http://www.ieee802.org/15/pub/TG2-Technical-Presentations.html/>, Zugriffsdatum: 11.03.2018

### 3. Ethernet im industriellen Bereich

standardisiert.

- **Zigbee:** ZigBee wurde für den Einsatz in drahtlosen Netzwerken mit geringen Datenaufkommen und mit eingeschränkter Energieversorgung (Batterie) entwickelt. Am häufigsten wird Zigbee in der Gebäudeautomatisierung eingesetzt. ZigBee ist im IEEE 802.15.4 Standard spezifiziert.
- **Mobilfunk:** Die Kommunikation erfolgt über die Mobilnetze der Telekommunikationsanbieter. Es werden die internationalen Mobilfunk-Standards verwendet (GPRS, HSPA, EDGE, UMTS, LTE). Die Reichweite ist global.
- **WirelessHART:** Dient zur drahtlosen Vernetzung von *Highway Addressable Remote Transducer* (HART)-Feldgeräten in der Prozessindustrie und ist nach IEEE 802.15.4 standardisiert. WirelessHART zeichnet sich durch eine hohe Zuverlässigkeit, sehr geringe Energieaufnahme und eine extrem abhör- und manipulationssichere Übertragung aus[[24]].

## 4. Bussysteme im Fahrzeug

Dieses Kapitel soll einen Einblick in die am häufigsten eingesetzten Bussysteme im Kraftfahrzeugen und deren Eigenschaften bieten.

### 4.1. CAN-Bus

Der *Controller Area Network* (CAN)-Bus wurde in den 80er Jahren von der Firma Bosch entwickelt und 1991 als erstes Bussystem in einem Fahrzeug eingesetzt[[7]]. Derzeit ist CAN das am meisten eingesetzte Fahrzeug-Bussystem. Die CAN Protokolle 2.0A und 2.0B, mit dem *29 bit Identifier*, stellen bis heute die Basis aller CAN-Implementierungen dar.

Mit ISO 11898- und SAE J2284-Standard wurde das Protokoll für die Anwendung im PKW und mit dem SAE J1939-Standard für die Anwendung im NKW international standardisiert (Abb. 4.1). Ab 2008 ist CAN die einzig zugelassene Schnittstelle bei Neufahrzeugen für Diagnose-Tester bei der *On Board Diagnose* (OBD)[[25]].

ISO 11898 – 1	<b>Data Link Layer</b> , entspricht den Bosch Spezifikationen CAN 2.0A und CAN 2.0B
ISO 11898 – 2, 5, 6 ISO 11898 – 3	<b>Physical Layer</b> für High Speed CAN ... und Low Speed CAN
ISO 11898 – 4	Erweiterung des Data Link Layer für zeitgesteuerte Kommunikation ( <b>Time Triggered CAN</b> )

Abbildung 4.1.: Die wichtigsten CAN-Standards[[25]]

Da hohe Stückzahlen an CAN-Controllern im Automobilbereich produziert werden sind diese preisgünstiger als die meisten ASICs für die in der Automatisierungstechnik verbreiteten Feldbusse (z.B. Profibus). Daher wird CAN auch in industriellen Anwendungen als Sensor-Aktor-Bus verwendet (*CAN in Automation*)[[25]].

Die gesamte Botschafts-Übertragung einschließlich Fehlerüberprüfung erfolgt beim CAN-Protokoll durch den CAN-Controller, welcher heute standardmäßig als *On-Chip-Modul* auf dem Mikrocontroller implementiert ist. Die physikalische Kopplung an die Busleitungen erfolgt durch einen separaten *CAN-Transceiver* (siehe Abb. 4.2). Alle neueren CAN-Controller sind CAN 2.0A- und CAN 2.0B-fähig.

## 4. Bussysteme im Fahrzeug

Um ein deterministisches Übertragungsverhalten zu ermöglichen muss ein synchronisierter Buszugriff erfolgen, bei welchem dem Bus-Teilnehmer vorgegeben wird, in welchem Zeitfenster er auf den Bus zugreifen darf. Deswegen wurde das Time-Triggered-CAN (TTCAN) in ISO11898-4 standardisiert, welches das *Time Division Multiple Access* (TDMA)-Buszugriffsverfahren verwendet. TTCAN ermöglicht die deterministische Übertragung von Nachrichten, erlaubt aber keine höheren Bitraten als der ursprüngliche CAN-Bus. TTCAN hat sich daher bis jetzt nicht durchgesetzt und soll mittelfristig in der Fahrwerks- und Antriebsstrang-Elektronik durch FlexRay ersetzt werden[[25]].

### 4.1.1. Physical Layer

CAN ist ein Bitstrom-orientiertes und auf der Linien-Topologie aufgebautes Bussystem, welches eine maximale Übertragungsrate von 1 Mbit/s ermöglicht. Für den Buszugriff verwendet CAN das *Carrier Sense Multiple Access/Collision Resolution* (CSMA/CR)-Verfahren. Die Fehlererkennung erfordert eine Reaktion aller Steuergeräte innerhalb einer Bitzeit und verkürzt damit die Buslänge desto höher die Bitrate wird[[25]].

Das verwendete Codierungsverfahren im CAN ist die *Non-Return-to-Zero*-Codierung. Das Verfahren hat den Nachteil, dass der Verlust an Synchronisationsinformation durch Bitstufing ausgeglichen wird[[27]].

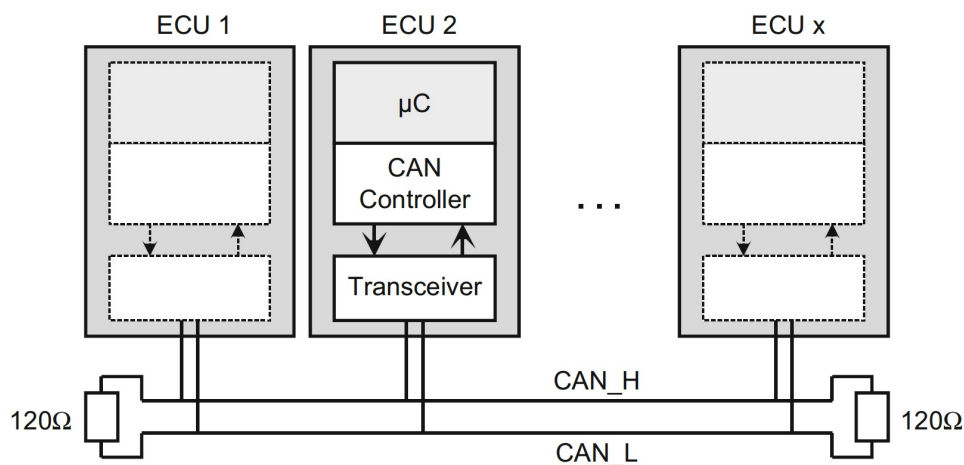


Abbildung 4.2.: High-Speed-CAN-Netzwerk mit Abschlusswiderständen[[25]].

Für Bitraten bis 250 kbit/s im ISO 11898-2 (*High Speed CAN*, Class C)-Anhang wird die Verwendung einer verdrehten Zwei-Draht-Leitung als Linien-Bus mit Stichleitungen von maximal 30 cm zu den einzelnen Bus-Teilnehmern definiert. Der Bus muss an beiden Enden der Zwei-Draht-Leitung mit einem typischerweise 120 Ω Widerstand abgeschlossen werden (siehe Abb. 4.2).

Der Signalhub des Differenzspannungs-Signals liegt bei ungefähr 2 V (Abb. 4.3). Für das Arbitrierungsverfahren im CAN-Protokoll ist eine Differenzierung der Buspegel in *dominant* und *rezessiv* notwendig. Bei einer Unterbrechung oder einem Kurzschluss einer Ader der Zwei-Draht-Leitung fällt der Bus komplett aus. In PKWs werden *High-Speed-CAN*-Busse mit Bitraten von 500 kbit/s im Antriebsstrang-Bereich und bei Nutzfahrzeugen mit Bitraten von 250 kbit/s eingesetzt. In den SAE-Normen für CAN in PKW (SAE J2284) und LKW (SAE J1939) sind diese Normen standardisiert[[25]].

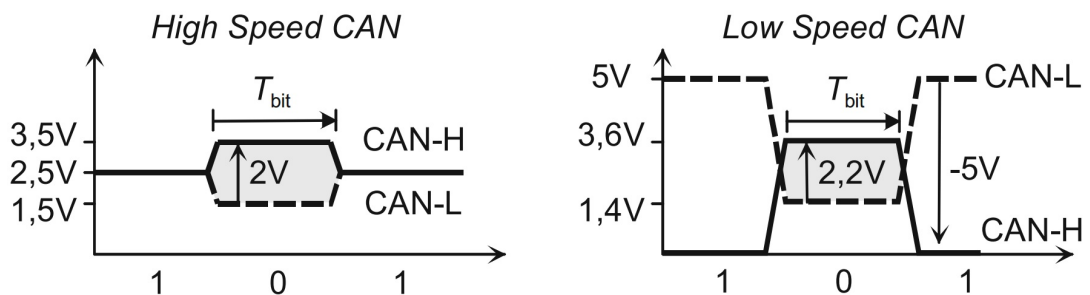


Abbildung 4.3.: Signalpegel bei High- und Low-Speed-CAN nach ISO 11898-2 und -3[[25]].

Im Anhang ISO 11898-3 wird für Bitraten bis 125 kbit/s (*Low Speed CAN*, Class B), die z.B. für Anwendungen in der Karosserie-Elektronik eingesetzt werden, auch eine Zwei-Draht-Leitung definiert. Der Bus darf entsprechend länger sein aufgrund der niedrigeren Bitrate. Es entfallen die Busabschluss-Widerstände und die Begrenzung auf kurze Stichleitungen. Der Signalhub des Differenzsignals ist größer als beim *High Speed CAN* (siehe Abb. 4.3). Auch bei Kurzschluss oder Kabelbruch einer Ader bleibt der Bus funktionsfähig[[25]].

#### 4.1.2. Fehlererkennung und -behandlung

Das CAN-Protokoll enthält fünf verschiedene Fehlererkennungs-Verfahren, welche für eine hohe Übertragungszuverlässigkeit sorgen. In der Tabelle 4.1 werden diese Fehlermechanismen beschrieben.

Wenn einer der beschriebenen Fehlern von einem Bus-Teilnehmer erkannt wird, initiiert dieser ein Fehlertelegramm. Dieses Telegramm wird als *Fehlerflag* bezeichnet, welches aus sechs dominanten Bits besteht und die übertragene Nachricht zerstört. Damit verstößt das Fehlerflag bewusst gegen die Bitstuffing-Regel und veranlasst den Sender die Nachricht erneut zu versenden[[27]].

## 4. Bussysteme im Fahrzeug

Tabelle 4.1.: Fehlermechanismen beim CAN-Bus[[27]]

Mechanismus	Erläuterung
Bitmonitoring	Der sendende Knoten prüft, ob der zur Sendung beabsichtigte Pegel auch auf dem Bus „erscheint“.
Überwachung des Telegrammformats	Jeder Netzknoten überwacht, ob die auf dem Bus gesendete Botschaft Formfehler enthält.
Zyklische Block-sicherung (CRC)	Bei diesem Verfahren wird aus Botschaftsbeginn, Arbitrierungsfeld, Steuerfeld und Nutzdaten eine Prüfsequenz durch Polynomdivision gemäß dem CRC-Verfahren gebildet. Diese Sequenz wird empfangsseitig ebenfalls gebildet und durch den Empfänger mit der übertragenen Prüfsequenz verglichen.
Überwachung Acknowledgement	Der Sender einer Botschaft erwartet die Bestätigung des fehlerfreien Empfangs durch Aufschaltung eines dominanten Pegels im ACK-Feld durch die Empfänger. Bleibt die Bestätigung aus, geht der Sender davon aus, dass ein Fehler aufgetreten ist.
Überwachung Bitstuffing	Alle Busteilnehmer überwachen die Einhaltung der Bitstuffing-Regel.

### 4.1.3. Nachrichtenformat

Die gesamte Botschafts-Übertragung erfolgt beim CAN-Protokoll durch den CAN-Controller. Jeder Teilnehmer sendet Broadcast-Nachrichten ohne Ziel- oder Quelladresse. Für die Erkennung werden Botschaften mit einem *Message Identifier* markiert. Es muss keine Verbindung aufgebaut werden.

Jeder Teilnehmer am Bus empfängt die Nachricht und entscheidet anhand des *Message Identifier* ob er diese verarbeitet oder ignoriert. Beim CAN 2.0A war der *Message Identifier* 11 bit, in der zweiten CAN-Generation (CAN 2.0B) 29 bit lang. Es werden noch drei Steuerbits im *Message Identifier* mitverschickt[[25]]. Der Botschaftsaufbau wird in der Abb. 4.6 dargestellt.

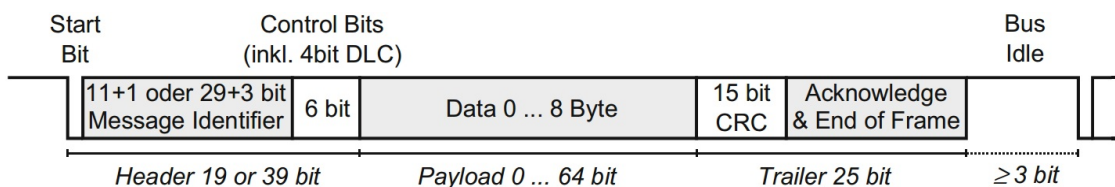


Abbildung 4.4.: Botschaftsaufbau des CAN-Standards[[25]].

CAN verwendet das CSMA/CR Buszugriffsverfahren, bei welchem jeder Teilnehmer senden kann, sobald der Bus für mindestens 3-Bitzeiten frei ist. Der *Message Identifier* kennzeichnet auch die Priorität einer Botschaft, wo eine niedrigere Zahl eine höhere Priorität bedeutet. Kommt es zu einer Kollision, so setzt sich die Botschaft mit dem niedrigeren *Message Identifier* durch.

Eine Botschaft kann Nutzdaten (*Payload*) von 0 bis 8 Byte enthalten. Diese Zahl steht im *Data Length Code* (DLC)-Feld innerhalb des *Control Bits*. Eine 15 bit lange *Cyclic Redundancy Check*-Prüfsumme wird zur Fehlererkennung mitverschickt. Die Empfänger synchronisieren ihren Bittakt-Generator über das Startbit mit dem Sender und werden durch Stuff-Bits nachsynchronisiert.

Die CAN-Controller der Teilnehmer am Bus überprüfen das Nachrichtenformat und die Prüfsumme und senden eine Empfangsbestätigung oder Fehlermeldung (*Error Frame*) innerhalb des *Acknowledge* und *End of Frame*-Feldes. Die empfangenen Daten werden bei einer Fehlermeldung vom Empfänger ignoriert[[25]].

#### 4.1.4. CAN FD

Mit CAN sind Übertragungsraten bis zu 1 Mbit möglich, was auch in den meisten CAN basierten Anwendungen ausreicht. Trotzdem gibt es Einzelfälle wo der Bedarf nach einer höheren Übertragungsrate da ist[[2]]. Deshalb wurde von Bosch die *CAN with Flexible Data-Rate* (CAN FD) Lösung in der Version 1.0<sup>1</sup> spezifiziert, mit welcher die Datenrate von CAN Systemen gesteigert wurde.

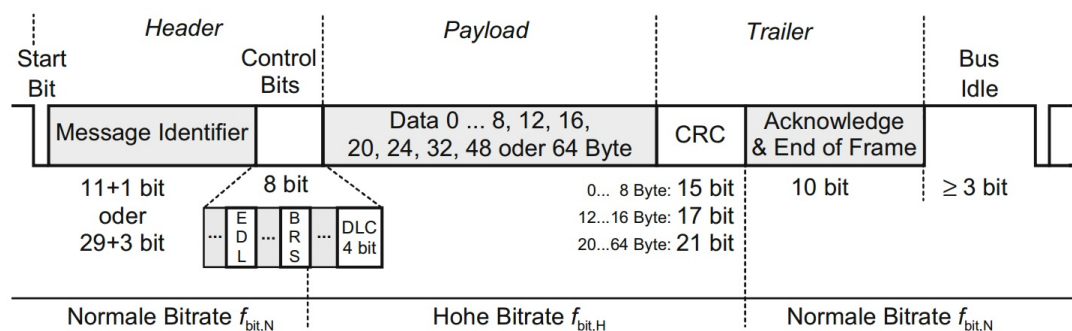


Abbildung 4.5.: Botschaftsaufbau des CAN FD[[25]]

Diese Steigerung der Übertragungsrate erfolgt durch längere Botschaften und einen höheren Bittakt. Der Botschaftsaufbau des CAN FD wird in der Abb. 4.5 dargestellt.

<sup>1</sup>Robert Bosch GmbH: „CAN with Flexible Data-Rate, Specification“, Version 1.0

#### 4. Bussysteme im Fahrzeug

Das *Payload* einer CAN-Nachricht wird von 8 auf 64 Byte erhöht. Damit können CAN FD-Nachrichten bis maximal 64 Byte Nutzdaten enthalten. Die Längenangabe *DLC* bleibt gleich, aber die CRC-Prüfsumme im Trailer muss bei jeder Botschaft angepasst werden. Im *Header* wird das *Extended Data Length* (EDL)-Bit eingeführt, welches ermöglicht eine CAN FD-Nachricht von einer normalen CAN-Nachricht zu unterscheiden.

Die Bitrate bei CAN wird durch zwei Faktoren begrenzt. Die bitweise-Arbitrierung des *Message Identifiers* im Nachrichten-Header zur Kollisionserkennung sowie durch das *Acknowledge Bit* zur Bestätigung des korrekten Empfangs im Nachrichten-Trailer.

Bei CAN FD wird die Bitrate in dem Nutzdatenfeld und der CRC-Prüfsumme deutlich erhöht. Für den Physical Layer aus ISO 11898-2 werden Werte um 4 Mbit/s erwähnt. Die Bitratenumschaltung ist optional und wird durch den *Bit Rate Switch* (BRS) im *Control-Feld* des Headers gesetzt (siehe Abb. 4.5). Am Ende des CRC-Felds wird auf die normale Bitrate zurückgeschaltet[[25]].



## 4.2. LIN-Bus

Local Interconnect Network (LIN) wurde Ende der 90er Jahre entwickelt, um eine kostengünstige Alternative zum Low-Speed-CAN Bussystemen zu bieten. Der LIN-Bus wird für die Vernetzung von einfachen Sensoren und Aktuatoren verwendet, wo die Bandbreite und Eigenschaften von CAN nicht erforderlich sind, z.B. für die Tür-, Sitz- oder Schiebedach-Elektronik[[9]].

Die physikalische Schicht besteht aus einer 12 V Eindrahtleitung nach ISO 9141<sup>2</sup>, wo der Buspegel direkt aus dem Bordnetz abgeleitet wird. Es sind Datenraten bis 20 kbit/s möglich. Üblich ist eine Begrenzung auf maximal 16 Busteilnehmer. Die aktuelle Version 2.1 wurde 2006 veröffentlicht, währenddessen der neue Standard 2.2A[[22]] bei der ISO (International Organization for Standardization) zur Billigung vorliegt.

### 4.2.1. Eigenschaften und Buszugriffsverfahren

Der LIN-Bus basiert im Gegensatz zu CAN, bei welchem jeder Knoten gleichberechtigt ist, auf einen Master- und mehreren Slave-Knoten. Der Zugriff auf den Bus wird durch den Master gesteuert. Der Master-Knoten fungiert als *Gateway* zu einem übergeordneten CAN-Netz (Abb. 4.6).

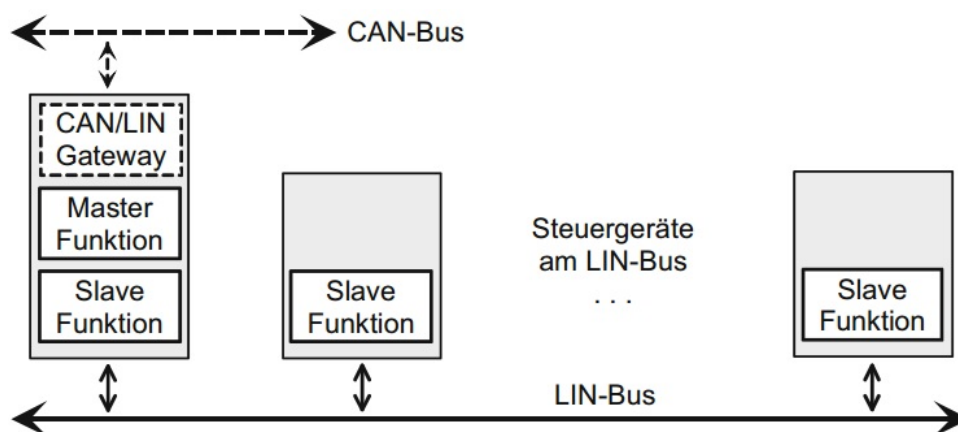


Abbildung 4.6.: Systemtopologie eines LIN-Bussystems[[25]].

<sup>2</sup>Norm ISO 9141: Straßenfahrzeuge; Diagnosesysteme; Anforderungen für den Austausch digitaler Informationen, 1992

#### 4. Bussysteme im Fahrzeug

Der Kommunikationsablauf auf dem Bus wird über *Tasks* abgewickelt, welche die Nachrichtenanteile repräsentieren die zwischen Master- und Slave-Knoten ausgetauscht werden. Das LIN-Protokoll regelt den Buszugriff mit dem Master-Slave-Verfahren. Im Master wird der Kommunikationsablauf in Form von *Schedule Tables* abgelegt, welcher durch das Versenden von *Headern* durch den Master-Task festgelegt wird. Der Slave-Task mit dem gleichen Identifier wie im gesendeten Header reagiert mit dem *Response-Anteil* der Nachricht (Abb. 4.7)[[27]].

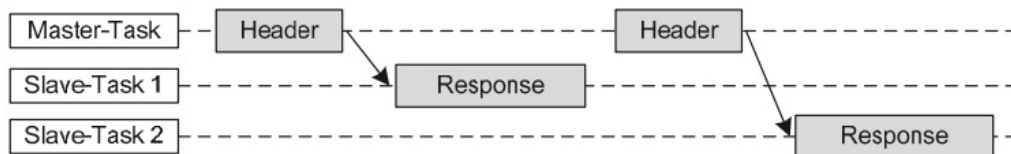


Abbildung 4.7.: LIN-Kommunikationsablauf[[22]].

Der Master übernimmt das Fehler-Management und überwacht die Slaves mit Hilfe eines definierten *Status-Bits*. Hingegen generieren die Slaves eigene interne Status-Bits.

#### 4.2.2. Nachrichtenformat

Eine LIN-Botschaft besteht wie in Abb. 4.8 aus dem Kopf (Header), welcher durch das *Break-Field* eingeleitet wird und so den Nachrichtenbeginn signalisiert. Anschließend werden die *Synchronisations-Bits* und der Nachrichten-Identifier gesendet. Das *Protected-Identifier-Field* besteht aus dem eigentlichen Identifier und zwei *Parity-Bits*. Nach dem Identifier wird ein Zwischenraum (*Response Space*) eingefügt, nach welchem die Daten (maximal 8 Byte) übertragen werden. Die Sicherung der Daten erfolgt durch eine Prüfsumme[[27]].

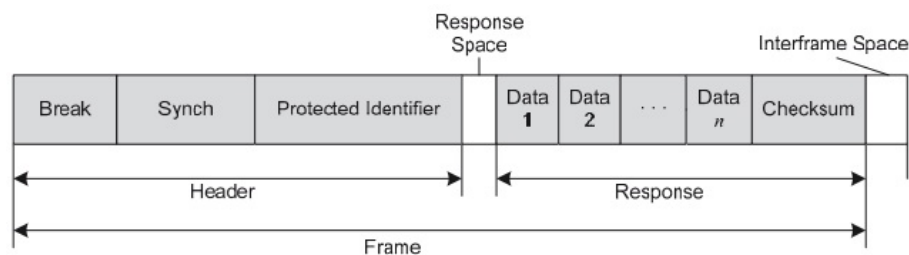


Abbildung 4.8.: LIN-Botschaftsformat[[22]]

Das LIN-Protokoll ermöglicht neben dem Standard-Nachrichtentyp, der zeitgesteuert übertragen wird, noch periodisches, ereignisgesteuertes oder sporadisches Sendeverhalten. Näheres dazu findet man unter [[22]].

## 4.3. FlexRay

FlexRay ist ein Bussystem, welches für die Steuerungs- und Regelungstechnik im Automobilbereich entwickelt wurde, insbesondere für sicherheitskritische Anwendungen. Es ist heute als *ISO 17458* standardisiert[[26]].

Ziel von FlexRay ist ein System mit hoher Bitrate (10 Mbit/s) anzubieten, das deterministisch und fehlertolerant arbeitet und dabei flexibel in Anwendungen und Erweiterungen ist. FlexRay wird insbesondere in Powertrain-Systemen (Antrieb) und Sicherheits-Systemen (X-by-Wire) eingesetzt.

Außerdem bietet FlexRay für deterministische Übertragungen einen zeitgesteuerten Buszugriff an, welcher für weniger hohe Anforderungen an die Übertragungsraten eine möglichst effektive Ausnutzung der Übertragungskapazität gewährleistet. Um diese Verfahren zu kombinieren, wird die Kommunikation in Zyklen abgewickelt[[28]].

In derzeitigen Fahrzeugen auf dem Markt wird FlexRay eher als Ergänzung denn als Ersatz für CAN eingesetzt.

### 4.3.1. Bus-Topologien

FlexRay bietet ein- und zweikanalige Systeme in Linien-Struktur, wie auch in Stern-Struktur (Abb. 4.9) an. Derzeit werden alle verfügbaren Kommunikationscontroller zweikanalig ausgeführt und als Grundtopologie wird die Stern-Struktur festgelegt.

Beim Linien-Bus und in der Stern-Struktur mit passivem Sternpunkt beträgt der maximale Abstand zwischen den entferntesten Busteilnehmern 24 m. Wegen den möglichen schlechten elektrischen Eigenschaften sind Stern-Strukturen mit passivem Sternpunkt nur bei kurzen Verbindungen oder niedrigen Übertragungsraten sinnvoll[[25]].

Der Sternpunkt sollte mit einem aktiven Sternkoppler ausgeführt werden, welcher als bidirektionaler *Transceiver* und *Repeater* die einzelnen Verbindungen elektrisch entkoppelt und die empfangenen Botschaften an alle angeschlossenen Steuergeräte weiterleitet. Die maximale Entfernung zwischen zwei Steuergeräten lässt sich durch die Zusammenschaltung zweier Sternpunkte auf 72 m steigern.

## 4. Bussysteme im Fahrzeug

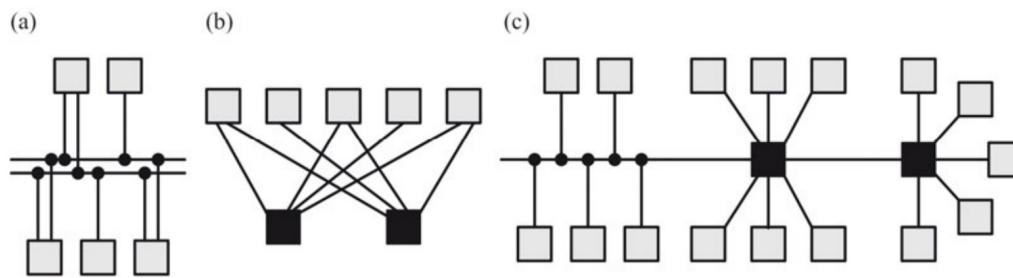


Abbildung 4.9.: Beispiele für FlexRay-Topologien. (a) Zweikanalige Bus-Topologie. (b) Zweikanalige Stern-Topologie. (c) Einkanalige hybride Topologie. Die schwarzen Vierecke repräsentieren die Stern-Koppler. Für andere Ausführungen siehe [[31]]

### 4.3.2. Physical Layer

Bei FlexRay ist die maximale Übertragungsrate derzeit 10 Mbit/s, welche sich aber nur in der Struktur mit aktivem Sternpunkt erreichen lässt. In der Linien-Struktur und bei passivem Sternpunkt sind 2,5 bis 5 Mbit/s möglich [[25]].

Wenn das System zweikanalig ausgeführt wird, kann der Busteilnehmer an beide oder nur an einen der beiden Kanäle angeschlossen werden (Abb. 4.9 a). Ein Steuergerät, das nur am Kanal X angeschlossen ist, kann nicht mit einem Steuergerät kommunizieren, welches nur am Kanal Y angeschlossen ist. Bei sicherheitskritischen Botschaften kann der zweite Kanal redundant verwendet werden, indem das Steuergerät eine Nachricht gleichzeitig über beide Kanäle verschickt.

Bei zweikanaligen Systemen ist eine gemischte (hybride (Abb. 4.9 c)) Topologie möglich, wo z.B. Kanal X als Stern und Kanal Y als Linienbus aufgebaut wird.

Die Verbindungen werden als geschirmte oder verdrehte Zweidraht-Leitungen, mit einem Wellenwiderstand von 80 bis 110  $\Omega$  ausgeführt. An den Leitungsenden sind wie bei CAN entsprechende Abschlusswiderstände notwendig. Es werden im Gegensatz zu CAN auch 0- und 1-Bits mit niederohmigen Differenzspannungs-Signalen übertragen (Abb. 4.10). Im Ruhezustand befinden sich beide Leitungen auf ungefähr 2,5 V [[25]].

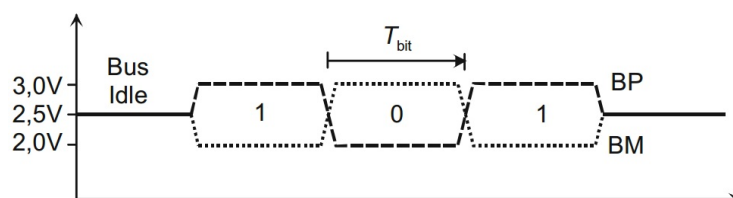


Abbildung 4.10.: Signalpegel auf den FlexRay-Leitungen BP und BM. [[25]]

### 4.3.3. Nachrichtenformat und Buszugriffsverfahren

Die FlexRay-Nachricht (*FlexRay-Frame*) ist in drei Segmente unterteilt (Abb. 4.11) und beginnt mit einem *Header*. Der Header enthält nach einigen Steuerbits im *Frame-ID-Feld* die Nummer des Zeitschlitzes (*Slot*) und die Länge der Nutzdaten welche in 16 Bit-Datenworten angegeben werden. Im letzten Zykluszähler-Feld (*Cycle*) des Headers befindet sich die Nummer des aktuellen Kommunikationszyklus, der mit Null beim Start des Systems initialisiert und mit jedem Kommunikationszyklus inkrementiert wird[[25]].

Im zweiten Nutzdatenworte (*Payload*)-Segment der Nachricht, welches maximal 254 *Byte* lang sein kann, werden die eigentlichen Daten übertragen. Über eine *Cyclic-Redundancy-Check* (CRC)-Prüfsumme im dritten *Trailer-Segment* wird die Datensicherung der Nachricht vorgenommen[[27]].

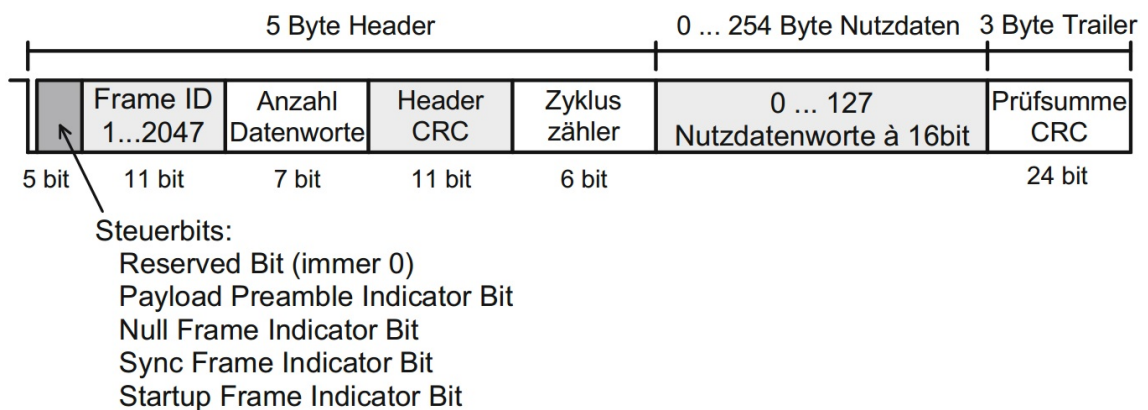


Abbildung 4.11.: FlexRay Nachrichtenformat[[25]]

Der periodisch wiederholte FlexRay-Kommunikationszyklus wird in ein *statisches* und ein optionales *dynamisches* Segment aufgeteilt (Abb. 4.12). Der Buszugriff bei FlexRay erfolgt über ein ähnliches Verfahren wie TTCAN, das sogenannte *Time Division Multiple Access* (TDMA)-Verfahren. Beim TDMA-Verfahren wird jeder Botschaft ein festes Zeitfenster zugeordnet, wodurch die gewünschten Echtzeiteigenschaften gewährleistet werden und so jede Nachricht zu einem fest definierten Zeitpunkt verschickt werden kann. Das ermöglicht eine deterministische Datenübertragung.

Im dynamischen Segment wird das FTDMA (auch *Mini-Slotting* bezeichnet)-Verfahren verwendet, welches für Nachrichten mit geringen Anforderungen an die Latenzzeiten geeignet ist. Die Steuerung des Buszugriffs erfolgt über einen Slotzähler[[27]].

#### 4. Bussysteme im Fahrzeug

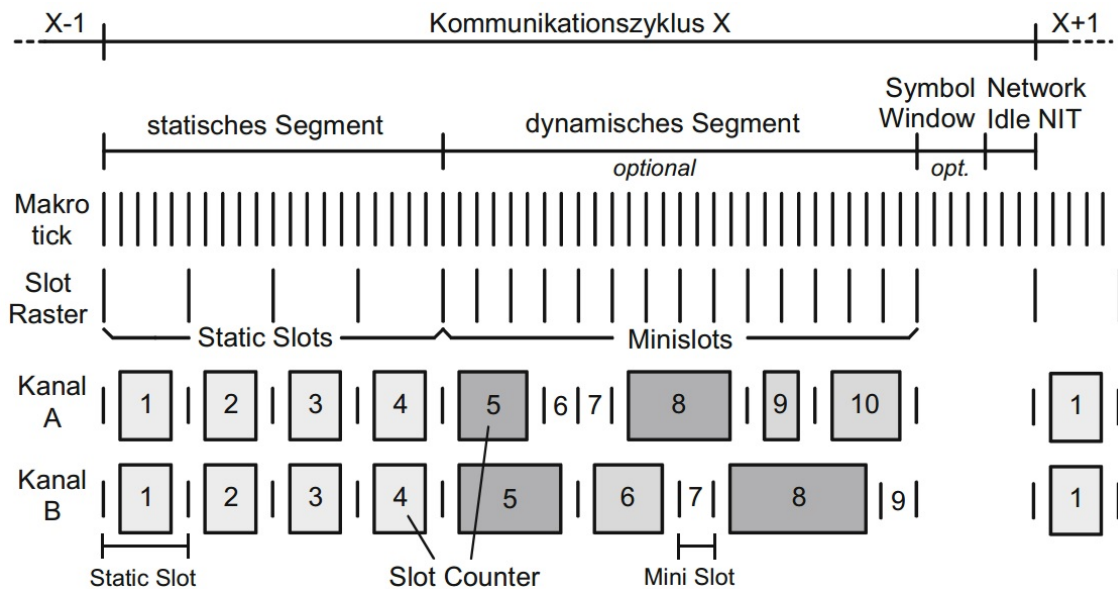


Abbildung 4.12.: Kommunikationsablauf bei FlexRay[[25]]

Um eine entsprechende Anzahl von *Sync-Frames* zu übertragen, welche für die Uhren-Synchronisation notwendig sind, beträgt die minimale Länge des statischen Segments zwei TDMA-Zeitfenster. Deswegen muss das *Network Idle Time* (NIT)-Feld (Abb. 4.12) vorhanden sein.

Da das dynamische-Segment und das Symbol-Fenster optional sind, ist es möglich die Aufteilung des Kommunikationszyklus an unterschiedliche Anwendungsfälle anzupassen. Das ermöglicht uns eine ausschließlich TDMA basierte Kommunikation aufzubauen, bei der das statische-Segment weiterhin zur Uhrensynchronisation genutzt werden kann. Das dynamische Segment kann daraufhin für eine eventgesteuerte Kommunikation mit priorisierter Reihenfolge und hoher Datenrate verwendet werden. In der Abb. 4.12 werden die Grundprinzipien der beiden Kommunikationen anhand eines zweikanaligen FlexRay-Systems veranschaulicht[[27]].

Ein *Bus-Guardian* kann sich in jedem Knoten oder in einem aktiven Sternpunkt des Netzwerks befinden und wird für die Überwachung der Kommunikation während des statischen-Segments benutzt. Bei Unstimmigkeiten kann der Bus-Guardian den Sender abschalten[[2]].

## 4.4. MOST

*Media Oriented System Transport* (MOST) wurde im Gegensatz zu den Bussystemen (CAN, LIN und FlexRay), die für Steuerung und Regelung konzipiert wurden (*Control Bus*), für Multimedia- und Telematik (*Infotainment Bus*) Anwendungen im Fahrzeug entwickelt.

Die Einsatzgebiete umfassen alle Multimedia-Anwendungen wie das Navigationssystem, Autoradio, Telekommunikation (Internet), Bordfernsehgerät usw. [[10]]. Für diese Anwendungen sind Anforderungen an Echtzeitverhalten und Übertragungssicherheit geringer, aber an Übertragungsbandbreiten desto höher.

Die MOST Spezifikation<sup>3</sup> umfasst die gesamten sieben ISO/OSI-Schichten. Die MOST-Kommunikationscontroller implementieren zusammen mit den Bus-Transceivern die ISO Protokollschichten 1 (Bitübertragungsschicht) und 2 (Datenübertragungsschicht), während die *MOST Network Services Layer 1 und 2* die Dienste der Schichten 3-7 abdecken [[25]].

Ethernet, welches schon für die Diagnoseschnittstelle und Kamerabasierte-Systeme im Fahrzeug vorhanden ist, könnte in Zukunft die Aufgaben von MOST im Infotainment-Bereich komplett übernehmen. Die *MOST Cooperation* versucht mit höheren Bandbreiten wie in *MOST150* (150 Mbit/s) und kostengünstigeren Busverkabelungen gegenzuhalten.

### 4.4.1. Physical Layer und Bus-Topologien

MOST25 ist die erste Generation der MOST-Systeme, die auf einem optischen Übertragungsmedium mit Kunststoff-Lichtwellenleitern (*Plastic Optical Fiber* (POF)) basiert. Die Daten werden *Manchester-codiert* und über das modulierte Lichtsignal übertragen. Der Lichtstrom wird im Steuergerät von einem optischen Transceiver (*Fiber Optical Transceiver* (FOT)) empfangen bzw. wieder ausgesendet.

Mit der zweiten Generation MOST50 wurde auch eine elektrische Variante mit einer verdrehten Zweidraht-Leitung eingeführt. MOST150 arbeitet auf optischer Basis, wobei es auch eine elektrische Variante gibt, die aber im Vergleich zu MOST50 teure Koaxialkabel verwendet.

---

<sup>3</sup><https://www.mostcooperation.com/publications/specifications-organizational-procedures/>, Zugriffsdatum: 11.03.2018

## 4. Bussysteme im Fahrzeug

Die am häufigsten eingesetzte MOST-Bus-Topologie ist eine logische Ring-Struktur mit maximal 64 Steuergeräten (Abb. 4.13)[[25]]. Ein MOST-System kann bis zu 64 Knoten haben. Ein Knoten wird als *Timing-Master* konfiguriert, welcher als Zeitbasis für alle Knoten im System fungiert.

Verbindungen zwischen Steuergeräten sind unidirektionale *Point-to-Point-Verbindungen*, das heißt in jedem Steuergerät wird das optische Signal regeneriert, bevor es zum nächsten Steuergerät weiter geschickt wird.

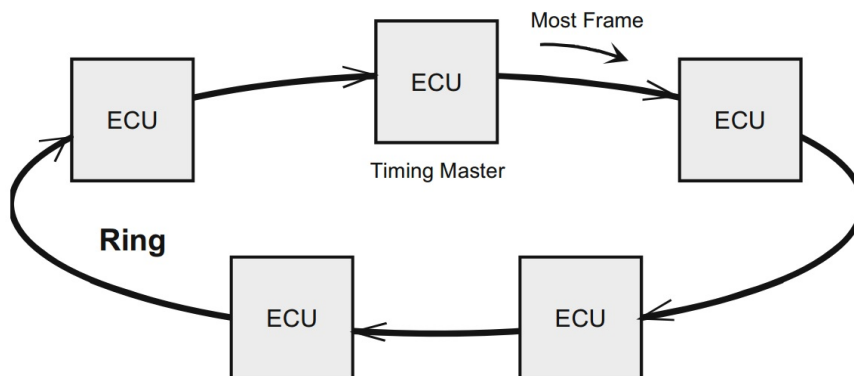


Abbildung 4.13.: MOST Ring-Struktur-Topologie[[25]]

Das MOST-Bussystem arbeitet in MOST25 mit einer Übertragungsrate von 25 Mbit/s, bei neueren Buscontrollern von 50 Mbit/s (*MOST50*) bis 150 Mbit/s (*MOST150*)[[25]].

### 4.4.2. Buszugriffsverfahren und Nachrichtenformat

Die MOST-Kommunikation ist in Blöcken organisiert, die aus 16 Botschaften (Frames) zusammengefasst werden (Abb. 4.14). Jede Botschaft durchläuft einmal die drei verschiedenen Kanäle. Die Botschaftslänge beträgt in der Regel  $22,67 \mu s$ , was der Abtastfrequenz von 44,1 kHz einer Audio-CD entspricht[[27]].



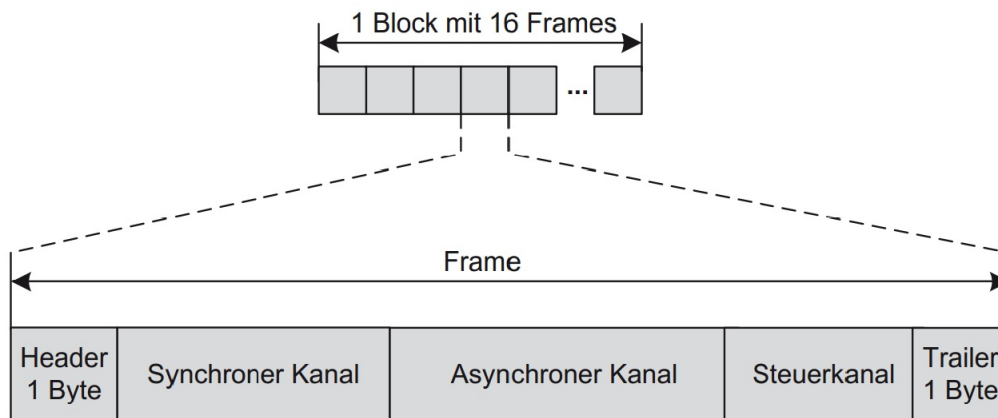


Abbildung 4.14.: MOST Nachrichtenformat[[27]]

Zur Übertragung von Echtzeit-Daten z.B. Videodaten, dient der *synchrone-Kanal*. Dieser Kanal verwendet das TDM-Verfahren, bei welchem die Datenströme mehrerer Sender mit niedriger Bitrate zu einem Datenstrom hoher Bitrate zusammengefasst werden.

Der *asynchrone-Kanal* versendet Daten-Pakete wie z.B. TCP/IP Pakete aus dem Navigationssystem. Im dritten Kanal werden Steuerdaten (*Control Channel*) z.B. Netzmanagement-Informationen übertragen.

Der *Boundary Descriptor* stellt die Grenze zwischen dem synchronen und asynchronen Kanal fest und wird im ersten Byte der Botschaft dem Header übertragen (Abb. 4.14). Die Gesamtlänge des Datenfeldes beträgt 60 Byte, davon mindestens 24 Byte für synchrone Daten und maximal 36 Byte für asynchrone Daten. Im letzten Segment des Frames, dem *Trailer*, werden Prüf- und Statusinformationen gesendet[[27]].

Die Tabelle 4.15 gibt eine kompakte Übersicht über die in diesem Kapitel beschriebenen Kfz-Bussysteme.

#### 4. Bussysteme im Fahrzeug

	CAN-C High-Speed-CAN	CAN-B Low-Speed-CAN	LIN			MOST-Bus	Flexray
<b>Bedeutung</b>	Controllier Area Network	Controller Area Network	Lokal Interconnect Network			Media Oriented Systems Transport	Kunstname
<b>Art des Busses</b>	konventioneller Bus	konventioneller Bus	konventioneller Bus			optischer Bus	konventioneller und optischer Bus
<b>Domäne</b>	Antriebsstrang	Komfort	Komfort			Multimedia und Infotainment	domänenübergreifender Einsatz
<b>Anwendung</b>	Vernetzung von Motormanagement, Getriebebesteuerung und ABS/ESP	Vernetzung von Karosserie- und Komfortelektronik	kostengünstige Erweiterung des CAN-Bus für einfache Anwendungen im Bereich Komfortelektronik			Übertragung von Steuer-, Audio- und Videodaten	ein Netzwerksystem zum Einsatz in sicherheitsrelevanten und einfacheren Anwendungen.
<b>meist angewandte Topologie</b>	Linearer Bus	Linearer Bus	Linearer Bus			Ring-Topologie	Stern-Topologie
<b>Datenübertragungsrate</b>	10 kbit/s bis 1Mbit/s	max. 125 kbit/s	max. 20 kbit/s			max. 22,5 Mbit/s	typ. 10 Mbit/s max. 20 Mbit/s
<b>max. Anzahl der Knoten</b>	10	24	16			64	theoretisch bis zu 2048 max. 22 pro passiven Bus/Stern
<b>Steuerungsmechanismus</b>	ereignisgesteuert	ereignisgesteuert	zeitgesteuert			zeit- und ereignisgesteuert	zeit- und ereignisgesteuert
<b>Busleitungen</b>	Kupferleitungen (twisted pair)	Kupferleitungen (twisted pair)	Kupferleitung (single wire)			Lichtleiter aus Kunststoff oder Glas	Kupferleitungen (twisted pair)
<b>Einsatz</b>	in allen Fahrzeugen	in allen Fahrzeugen	in allen Fahrzeugen			Oberklasse-Fz. europäischer Hersteller	Pilotanwendung
<b>Norm</b>	ISO 1198	ISO 11519-2	LIN-Konsortium			MOST Kooperation	Flexray-Konsortium
<b>SAE-Klassifizierung</b>	Klasse C	Klasse B	Klasse A			Mobile Media	Drive-By-Wire

Abbildung 4.15.: Übersicht der im Kfz eingesetzten Bussysteme[[28]]

## 5. Ethernet im Kraftfahrzeug

In diesem Kapitel wird die Motivation hinter der Einführung vom Ethernet ins Kraftfahrzeug beschrieben, sowie die schon vorhandenen Lösungen und die in der praktischen Arbeit eingesetzte BroadR-Reach® Technologie.

### 5.1. Motivation

Steigende Anzahl von Steuergeräten (derzeit bis zu hundert) und die zunehmende Vernetzung des Fahrzeugs intern und mit der Fahrzeugumgebung (Car2X), führen zu neuen Anforderungen an die Vernetzungs-Technologien im Fahrzeug (Abb. 5.1).

Dies war der Anlass um den seit Jahren aus der PC- und Industrie-Welt bewährten und erprobten Ethernet Standard ins Automobil einzuführen. Ethernet bietet neben den geringen Kosten auch eine hohe Bandbreite in der Onboard-Kommunikation als auch bei der Flashprogrammierung.

Zurzeit ist 100 Mbit/s im Fahrzeug Standard, währenddessen bewegt man sich im *Consumer-Bereich* (Standard-Ethernet) in Richtung 40 Gbit/s bis 100 Gbit/s, weshalb es viel Raum für Weiterentwicklung gibt. Im Unterschied zu anderen Bussystemen stehen bei Ethernet (z.B. BroadR-Reach®) 100 Mbit/s per Verbindung und Richtung zur Verfügung.

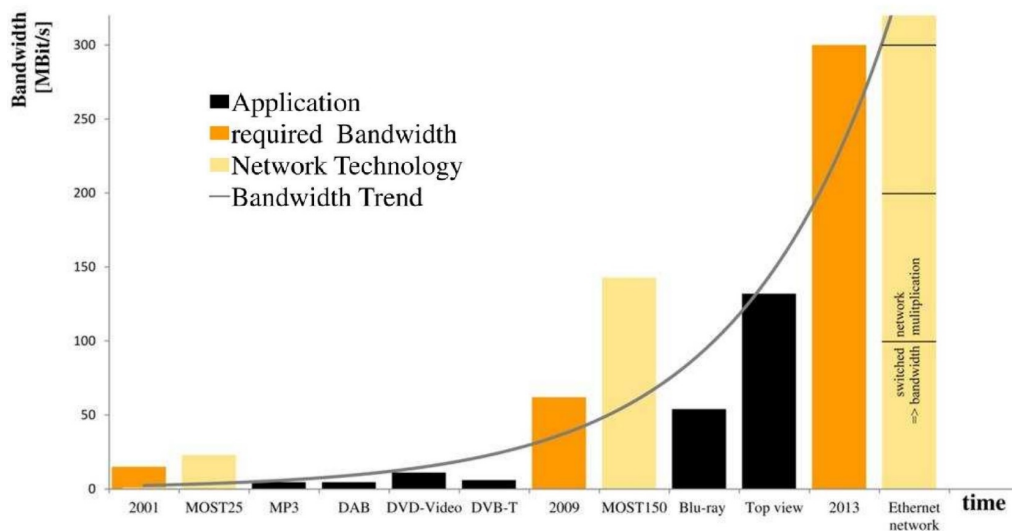


Abbildung 5.1.: Steigende Anforderungen an die Bandbreite treiben die stetige Optimierung der vorhandenen Technologien und die Einführung von neuen[[19]]

## 5. Ethernet im Kraftfahrzeug

Ethernet erleichtert den Einzug der *Consumer Electronic* (einfache *Plug & Play* Anbindung von Endgeräten und Applikationen) ins Fahrzeug. Mobile Endgeräte von Fahrzeuginsassen sollen Zugang zum Fahrzeugnetz bekommen und somit Multimedia (Video, Musik)- und Kommunikationsinhalte (Telefon, Mail, SMS) mit dem internen Fahrzeug-Multimediasystem nutzen können.

Da Ethernet eine bewährte und weit verbreitete Technologie ist, steht eine große Anzahl an verwendbaren Protokollen und Applikationen aus der PC- und Industrie-Ethernet-Welt zur Verfügung. Weiters sind viele Entwicklungs- und Diagnosewerkzeuge bereits vorhanden.

Vorhandene Bussysteme sind unflexibel und können mit der wachsenden Netzwerk-Komplexität nicht mithalten. Durch die Verwendung von Switches, bietet Ethernet eine einfache Skalierbarkeit des Netzwerks (*Switched Network*) an.

Es gibt mehrere proprietäre Standards für die Kommunikation in einem Fahrzeug, LIN, CAN, Flexray, MOST und LVDS. Jede Fahrzeugkomponente hat typischerweise eigene Kommunikationsanforderungen und eine eigene dedizierte Verkabelung.

Aufgrund dieser komplexen Verkabelung ist der Kabelbaum (Abb. 5.2) der dritt-höchste Kostenfaktor (hinter dem Motor und Chassis) und ebenfalls die dritt-schwerste Komponente im Fahrzeug[[17]].



Abbildung 5.2.: Ein typischer Kabelbaum im Auto[[12]]

Der Fahrzeug-Kabelbaum wird sich aus heterogenen Netzwerken, bestehend aus proprietären Protokollen wie MOST und CAN, in hierarchische homogene Ethernet-Netzwerke umwandeln (Abb. 5.3).

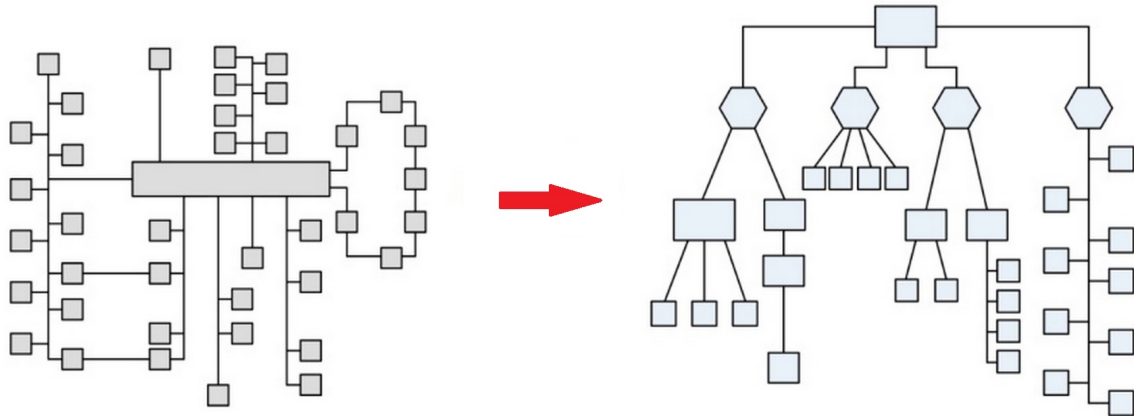


Abbildung 5.3.: Dank Ethernet wird die Umstellung vom heterogenen ins homogene Netzwerk gelingen[[30]]

Diese neue Fahrzeug-Netzwerkarchitektur hilft nicht nur die Kosten und das Gewicht zu verringern, sondern vereinfacht auch die Kommunikation zwischen den verschiedenen Systemen in- und außerhalb des Fahrzeugs[[17]].

### 5.1.1. Wieso wird Ethernet erst jetzt in Fahrzeugen eingesetzt?

Ethernet existiert schon über 25 Jahre, wurde aber von der Automobilindustrie bis jetzt nur außerhalb des Fahrzeugs eingesetzt (Schnittstelle für Diagnostik und *firmware updates*). In der Automobil Umgebung gelten andere Anforderungen als für Computer Netzwerke in Gebäuden (Störsicherheit, Gewicht, Umwelt-Einflüsse, Echtzeitfähigkeit, Kosten, Länge der Leitungen usw.)

Folgende Limitationen haben dessen Einstieg ins Auto hinausgezögert:

- Ethernet entsprach nicht den OEM EMV/RFI-Anforderungen für den Automobilmarkt. 100 Mbit/s (und höher) Ethernet hatte zu viel RF „noise“ und war anfällig für „alien“ Störungen durch andere Geräte im Auto.
- Ethernet konnte keine Latenz bis in den unteren Mikrosekundenbereich garantieren. Dies war erforderlich um die Kommunikation zu jedem Sensor/Steuergerät welcher eine schnelle Reaktionszeit erfordert, ersetzen zu können.

## 5. Ethernet im Kraftfahrzeug

- Es fehlte eine wirtschaftliche (kostengünstige) Verkabelungstechnologie. Da Gewicht und Platzmangel eine große Rolle spielen, wäre die herkömmliche Computer-Ethernet-Verkabelung zu groß und zu schwer gewesen.
- Ethernet ist in seiner Grundform trotz der hohen Übertragungsraten nicht echtzeitfähig und daher für Echtzeitanwendungen, wie beispielsweise Fahrer-Assistenzsysteme, ungeeignet. Deshalb mussten Echtzeitprotokolle entwickelt und in die relevanten Ethernet-Standards implementiert werden.
- Architektur der vorhandenen Steuergeräte musste auf Ethernet ausgelegt werden, was eine Erhöhung der Hardware- und Softwareanforderungen bedeutete. Weiters mussten die Test- und Entwicklungswerkzeuge für Automobiles-Ethernet angepasst werden.

Wegen dieser Einschränkungen von Standard-Ethernet musste die komplette physikalische Schicht neu entwickelt werden (BroadReach®).

### 5.2. Zukünftige Fahrzeugtechnologien

- **Vernetzte Fahrzeuge:** Aftermarket-Produkte, die das Fahrzeug über Mobilfunk oder WLAN mit dem Internet verbinden, sind schon seit längerer Zeit verfügbar. Diese Systeme werden nicht nur Funktionen wie Echtzeit-Verkehrsinformationen, Kommunikations- und Video-Streaming ermöglichen, sondern auch die Ferndiagnose und Firmware-Aktualisierungen (diese brauchen Zugang zum internen Netzwerk und Rechnern im Fahrzeug)[[17]].
- **Car2Car (C2C) Kommunikation:** Wird für die gegenseitige Koordination unter Fahrzeugen verwendet werden. Mit dem Wissen über Fahrzeugverhalten und Informationen über die Beziehung von Fahrzeugen untereinander sowie zu anderen Objekten können Sicherheitsabstände berechnet und Ausweichmanöver durchgeführt werden. Die *National Highway Traffic Safety Administration* schätzt, dass durch die Verwendung von C2C Technologien 79% der Auffahrunfälle vermieden werden können.
- **Augmented Reality:** Informationen über Objekte auf der Straße, wie beispielsweise die Distanz, werden auf die Windschutzscheibe projiziert. Gegebenenfalls kann das System dem Fahrer mittels einer Richtungsanzeige Anweisungen zur Umgehung der Gefahr geben.
- **Selbstfahrende (Autonome) Fahrzeuge:** Werden in der Lage sein, Passagiere und Lieferungen ohne Fahrer zum gewünschten Ort zu bringen. Die Selbstfahrenden Fahrzeuge der Firma Google sind in Kalifornien und Nevada bereits mehr als 300.000 Meilen auf öffentlichen Straßen und Autobahnen gefahren. Durch den Einsatz von Kame-



ras, Radar und Laser können die Autos Informationen schneller und zuverlässiger verarbeiten als Menschen. Diese Autos können durch *Platooning* die Straßen effizienter nutzen und in sehr engen Parkplätzen parken. Viele Branchen-Experten sind überzeugt, dass bis zum Jahr 2020 selbstfahrende Fahrzeuge verkauft und auf öffentlichen Straßen bewegt werden können[[17]].

### 5.3. Jetzige Automotive Ethernet Architektur

Heutzutage wird Ethernet von Automobilherstellern nicht nur für Diagnose oder Firmware-Updates genutzt, sondern zum Teil auch für kamerabasierte Fahrerassistenz-Systeme und im Bereich des Infotainments.

Dafür wird die Broad-Reach® Technologie ( (Abschn. 5.4.1)) verwendet, welche die strengen EMV/RFI-Anforderungen der Automobilindustrie erfüllt. Für alle sicherheitskritischen Anwendungen und den Antriebsstrang kommen weiterhin Flexray und CAN/CAN-FD zum Einsatz.

Die Umstellung auf Ethernet passiert zügig, da die konkurrierenden Technologien, welche die benötigte Bandbreite unterstützen (MOST und LVDS), teuer und proprietär sind[[17]]. Die Abb. 5.4 zeigt den heutigen Einsatz von automotivem Ethernet in die Fahrzeug-Netzwerkarchitektur.

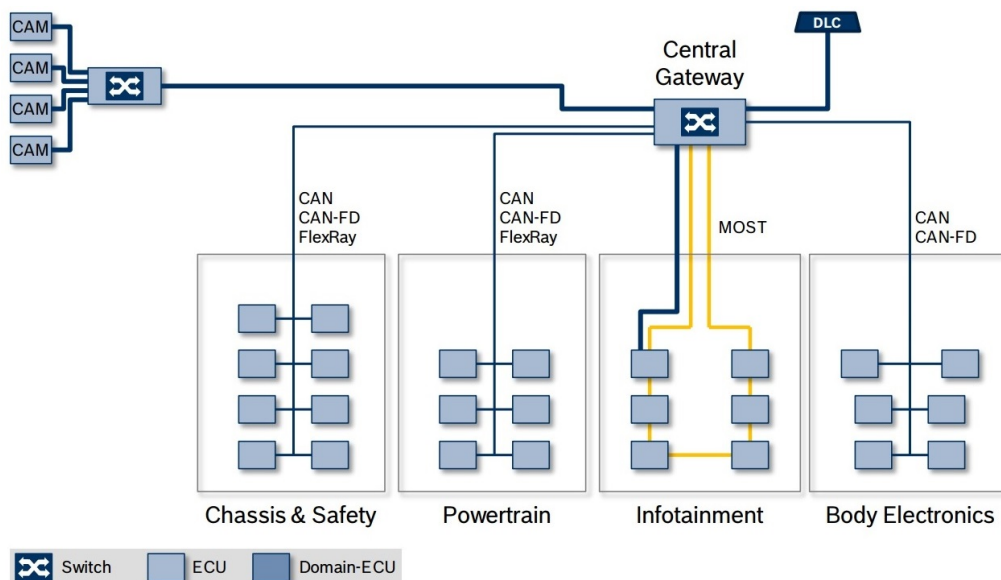


Abbildung 5.4.: Automotive Ethernet Architektur in 2015[[12]]

## 5. Ethernet im Kraftfahrzeug

### 5.4. OPEN Alliance SIG

Die *OPEN Alliance (One-Pair Ether-Net) Special Interest Group (SIG)*<sup>1</sup> ist ein Interessenverband, der im November 2011 von den Unternehmen NXP, Broadcom, Freescale und Harman gegründet worden ist. Das gemeinsame Ziel ist, die Ethernet-Technologie in der Automobilindustrie voranzutreiben und als offenen Standard für automotives Ethernet zu etablieren.

Es soll eine breite Basis an Anbietern aufgebaut werden, welche das automotive Ethernet weiterentwickeln und kompatible Produkte auf dem Markt bringen wollen. Weiters soll der entwickelte Standard Industrieanforderungen, wie z.B. Fahrzeugsicherheit, Infotainment und Komfort, verbessern. Dabei sollen sich die Netzwerkkomplexität und Verkabelungskosten signifikant verringern. Ethernet soll zunächst MOST und später auch FlexRay ergänzen bzw. ersetzen.

Bis jetzt haben sich schon etliche OEMs, Werkzeughersteller und Automobilzulieferer (z.B. BMW, Continental, Daimler, Bosch, Hyundai) diesem Konsortium angeschlossen. Die OPEN Alliance SIG zählt bis dato etwa 250 Mitglieder[[30]].

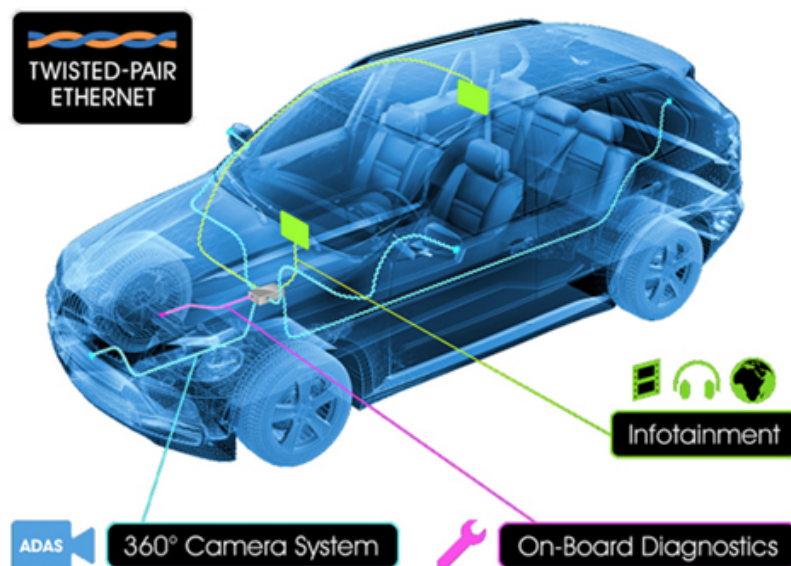


Abbildung 5.5.: One-Pair Ethernet BroadR-Reach® Technologie[[30]]

<sup>1</sup><http://www.opensig.org/>, Zugriffsdatum: 11.03.2018



Die OPEN Alliance SIG setzt auf die von Broadcom entwickelte BroadR-Reach® Ethernet Technologie (Abb. 5.5), welche eine 100 Mbit/s Übertragung über ungeschirmte *Single Twisted Pair Kabel* im Vollduplex-Modus ermöglicht. Derzeit gibt es nur eine begrenzte Anzahl an Lieferanten für PHY-Bausteine (Broadcom, Freescale und NXP). 100 Mbit/s BroadR-Reach® ist heute der De-Facto-Standard für Automotive-Ethernet[[30]].

#### 5.4.1. BroadR-Reach®

BroadR-Reach® ist eine neue Physikalische Schicht für Ethernet, welche von der Firma Broadcom für automotives Ethernet entwickelt wurde. Die BroadR-Reach® PHY-Bausteine erlauben derzeit eine Übertragungsrate von 100 Mbit/s im Vollduplex-Betrieb, über eine ungeschirmte, verdrehte, Zwei-Draht-Leitung (Abb. 5.7). Die OPEN Alliance SIG wurde gegründet, um BroadR-Reach® als offenen Standard für Automotive-Ethernet zu etablieren[[25]].

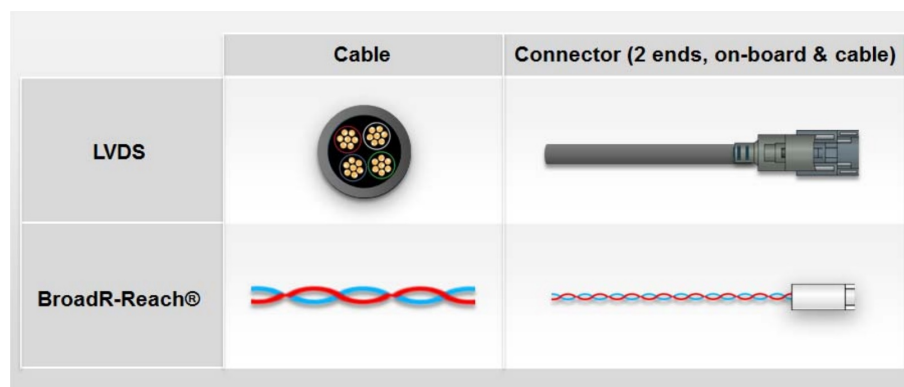


Abbildung 5.6.: Verkabelungsvergleich zwischen LVDS und BroadR-Reach®[[30]]

Die BroadR-Reach® Ethernet Technologie hat gegenüber anderen Breitband-Technologien wie z.B. LVDS deutliche Kostenvorteile, da sie mit einer ungeschirmten, verdrehten Zwei-Draht-Leitung arbeitet (Abb. 5.6). *Single-UTP-Kabel* sind nicht nur kostengünstiger und leichter als die geschirmten Kabel, sondern diese werden auch von bestehenden CAN- und LIN-Bussystemen verwendet. Damit sollen Gewichtseinsparungen bei der Verkabelung von 30% und Kosteneinsparungen von 80% möglich sein.

## 5. Ethernet im Kraftfahrzeug

### 5.4.1.1. Physical Layer

Die Übertragung erfolgt über eine ungeschirmte, verdrehte Zwei-Draht-Leitung (siehe Abb. 6.13). Die angegebenen Anforderungen an die Verkabelung und Steckverbinder sollen nicht höher als bei FlexRay sein.

Der *BroadReach® Physical Layer* (BR-PHY) unterstützt Standard-Ethernet-MAC Schnittstellen über das MII (IEEE-Standard 802.3 Paragraf 22) und nutzt Teile des IEEE 802.3 100BASE-TX- und 1000BASE-T-Standards. Wie man in Abb. 5.7 sieht, entspricht das MII (*Media Independent Interface*) dem Standard-Ethernet (siehe Abschn. 2.2), es ändert sich nur die physikalische Schicht.

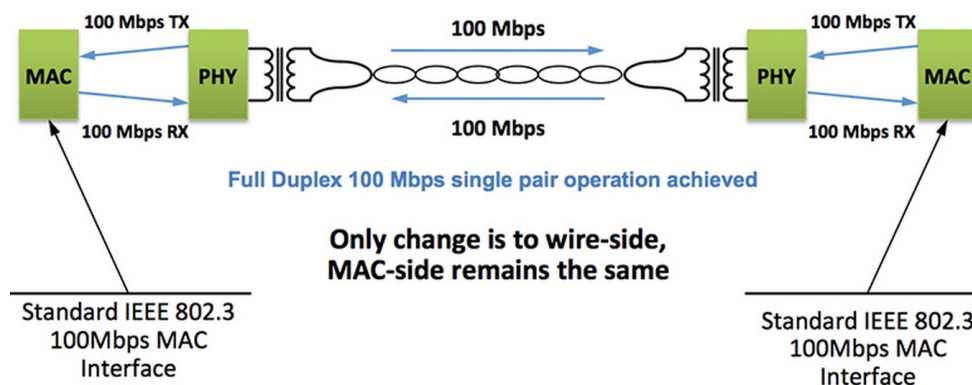


Abbildung 5.7.: 100 Mbit/s BroadReach® Technologie[[30]]

Der Transceiver (Physical Layer) vom BroadReach® arbeitet auf dem Prinzip von Gigabit-Ethernet, aber nur bei 100 MBit/s und nur auf einer Zwei-Draht-Leitung. Es wird die *Pulse Amplitude Modulation 3* (PAM3) Technologie verwendet, mittels welcher das Signal auf einen Träger aufmoduliert wird und die Daten als Kombination von Amplitude und Phase übertragen werden, statt eines Signalpegels pro Bitsignal (wie bei CAN, FlexRay und Standard-Ethernet). Auf diese Weise können mehrere Datenbit auf einmal übertragen werden[[29]]. Durch die Verwendung von *Echo Cancellers*, kann eine Bidirektionale Übertragung auf einem einzigen Paar stattfinden. In der Abb. 5.8 werden die Unterschiede zwischen physikalischen Ethernet-Schichten dargestellt.

Der *Physical Coding Sublayer* (BR-PCS) und *Physical Media Attachment Sublayer* (BR-PMA) bilden zusammen die BroadReach® physikalische Schicht.

BR-PCS sendet/empfängt Signale zu/vom MII. Es führt eine 4B3B Kodierung der empfangenen Signale durch und leitet diese zur weiteren Verarbeitung zum BR-PMA. BR-PCS empfängt 4-Bits am MII bei 25 MHz und wandelt diesen 4-Bit-Strom (25 MHz) zu einem 3-Bit-Strom bei  $33 \frac{1}{3}$  MHz um[[3]].

BR-PMA stellt die Schnittstelle zwischen BR-PCS und MDI dar. Es sendet/empfängt Signale zu/von BR-PCS auf die verdrehte Zwei-Draht-Leitung. BR-PMA verwendet die PAM3 Modulation und unterstützt *Link Management* und *BR-PHY Control functions*[[3]].

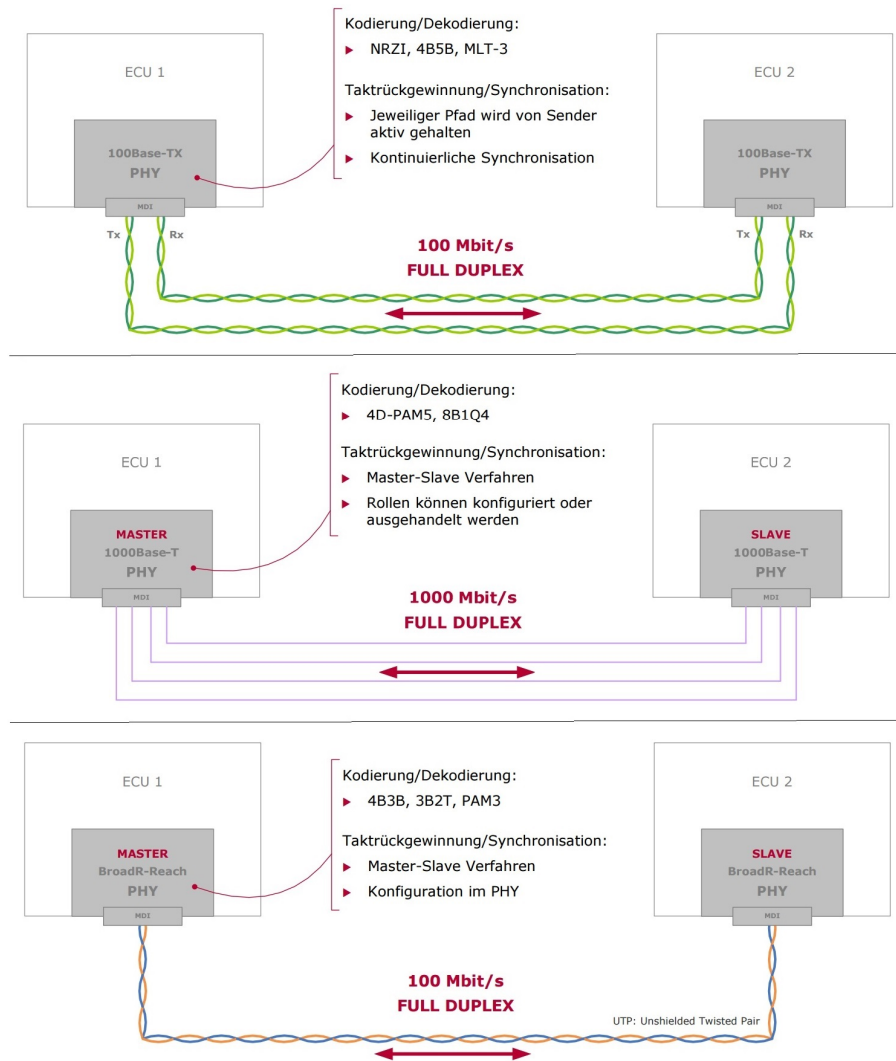


Abbildung 5.8.: Unterschiede zwischen den IEEE 100Base-TX, 1000Base-T und BroadR-Reach® physikalischen Schichten [[34]]

Durch das PAM Verfahren und die geringe Bandbreite (27 MHz vs. 62.5 MHz für 100 Base-T) erfüllt diese Norm die Automobil-EMV-Anforderungen (CISPR 25). Dies ist ein bedeutender Vorteil für die BroadR-Reach® Technologie.

Im Praktischen Teil (Kapitel 6) der Masterarbeit wurden die BroadR-Reach® BCM89810-Transceiver von Broadcom verwendet.

## 5. Ethernet im Kraftfahrzeug

### 5.5. Echtzeitanwendungen mit Automotive-Ethernet

Da im IEEE 802.3 Ethernet-Standard die zulässige Durchlaufverzögerung, wie auch die Reihenfolge bei der Weiterleitung von empfangenen Botschaften, nicht geregelt ist, ist das Ethernet in seiner Grundform nicht echtzeitfähig[[25]].

Um Echtzeitähnliche Anwendungen wie *Voice over IP* oder *Video Streams* trotzdem zu realisieren verwendet man Protokolle der höheren Schichten wie z.B. das *Real Time Transport Protocol* (RTP) nach RFC 3550. Diese Lösungen können das Problem der variablen Übertragungsverzögerung (Jitter) selbst nicht lösen, sondern nur kaschieren.

In geschlossenen Regelkreisen oder bei zeitkritischen Steuerungsdaten müssen alle Botschaften mit einer bekannten Maximalverzögerung übertragen werden. Botschaften, welche nicht übertragen werden, können ein Sicherheitsrisiko darstellen.

In der Industriesteuerung wird Ethernet schon seit einigen Jahren eingesetzt. Daher wurden mehrere Echtzeitlösungen wie ProfiNet, EtherCAT, EtherNet/IP, Modbus/TCP, Sercos III oder TTEthernet entwickelt. Da hinter vielen dieser Lösungen nur ein einzelner großer Automatisierungshersteller steht, fehlt ein einheitlicher Industriestandard[[25]].

Um einen kostengünstigen Industriestandard zu entwickeln, haben sich Consumer-Geräte- und Computerhersteller zusammengeschlossen und ein Konzept für das *Audio-Video Bridging* (AVB) entwickelt. Dieses ermöglicht eine Echtzeit-Übertragung von Audio- und Videodaten über den Ethernet *Data Link Layer*[[25]].

Der nächste Schritt wird die Verwendung von Ethernet in aktiven Sicherheits-Systemen und danach als zentrale Lösung (Backbone) im Fahrzeug sein. Für automotives Ethernet wird es entscheidend sein, ein einheitliches, kostengünstiges, echtzeitfähiges Konzept zu entwickeln, dass die strengen zukünftigen Anforderungen erfüllt.

### 5.5.1. Echtzeitfähigkeit mit AVB

Viele Fahrer-Assistenzsysteme sind auf die Daten von Kameras und anderen Sensoren innerhalb einer fixen Zeitspanne angewiesen. Anders als beim Schauen von Videos auf einem PC, in welchem durch Pufferung das unzuverlässige Timing des Netzwerks kompensiert wird, erfordern diese Audio-Video-Systeme sowohl kontrollierte Latenzen wie auch eine garantierte Bandbreite[[17]].

Ethernet ist in seiner Grundform, trotz der hohen Übertragungsraten, nicht echtzeitfähig und deshalb für Echtzeitanwendungen im Fahrzeug in dieser Form ungeeignet. Um eine kostengünstige Lösung zu finden, wurde vor über einem Jahrzehnt die *IEEE 802.3 Residential Ethernet Study Group* gegründet, welche später in die *IEEE 802.1 Audio/Video Bridging Task Group* geändert wurde.

Diese entwickelte ein Konzept für das *Audio-Video-Bridging* (AVB), das Echtzeitprotokolle in die relevanten Ethernet-Standards implementiert und so die Echtzeitübertragung von Audio- und Videodaten (Abb. 5.9) über den Ethernet *Data Link Layer* ermöglicht hat. AVB ist unter IEEE 802.1 standardisiert<sup>2</sup>.

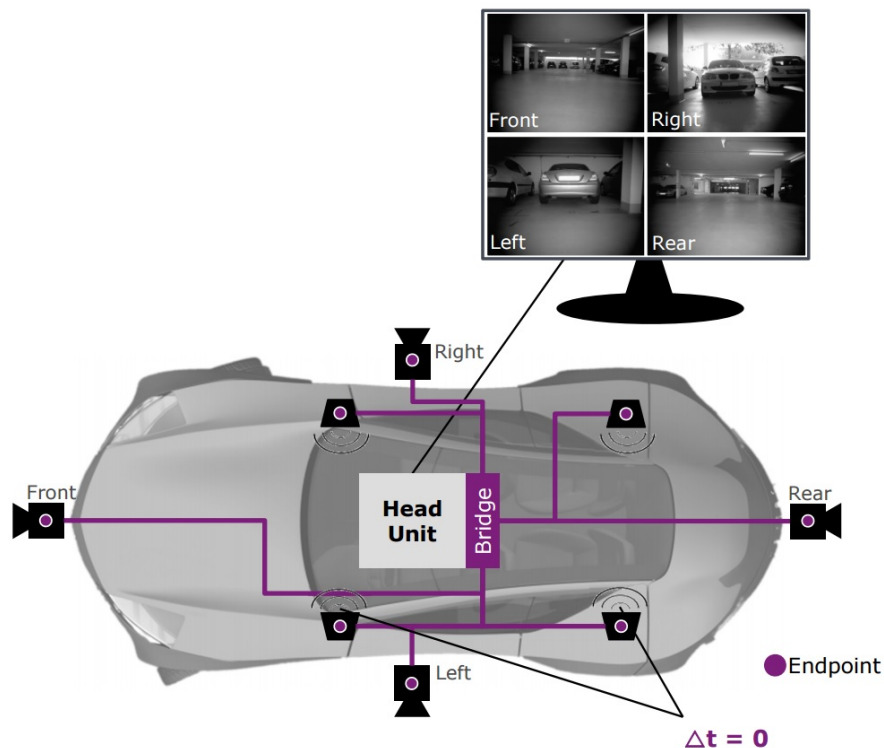


Abbildung 5.9.: Echtzeit-Übertragung von Audio- und Videodaten mit AVB[[34]]

<sup>2</sup><http://www.ieee802.org/1/>, Zugriffsdatum: 11.03.2018

## 5. Ethernet im Kraftfahrzeug

Die Entwickler von AVB haben sich in der *AVnu Alliance* organisiert, welcher sich mittlerweile auch die Automobilhersteller und Zulieferer der *OPEN Alliance* angeschlossen haben, um Ethernet vollständig echtzeit- und automobiltauglich zu machen[[25]].

AVB setzt sich aus folgenden Teilnormen zusammen:

- **IEEE 802.1BA:** Beschreibt, wie ein AVB-Netzwerk konfiguriert wird.
- **IEEE 802.1AS:** *Timing and Synchronization for Time-Sensitive Applications* ist ähnlich der IEEE-1588-Norm für Uhrensynchronisation und beschreibt ein Protokoll für Synchronisation von lokalen Uhren in einem Netzwerk.
- **IEEE 802.1Qat:** *Stream Reservation Protocol* oder Datenfluss-Reservierung, ist ein Protokoll welches die Reservierung von Bandbreite für sicherheitskritische Datenströme in einem Netzwerk ermöglicht.
- **IEEE 802.1Qav:** *Forwarding and Queuing Enhancements* definiert einen Mechanismus zur Behandlung von Echtzeitdaten in einem Ethernet-Switch[[4]].

Mit dem *Multiple Stream Reservation Protocol* (MSRP) können Übertragungskanäle (Streams) mit garantierter Bandbreite und maximaler Übertragungsverzögerung eingerichtet werden. MSRP reserviert 75% der Bandbreite für AVB-Nachrichten und strebt eine Gesamtlatenz von unter 2 ms, bei bis zu sieben Teilstrecken (Hops), an[[25]].

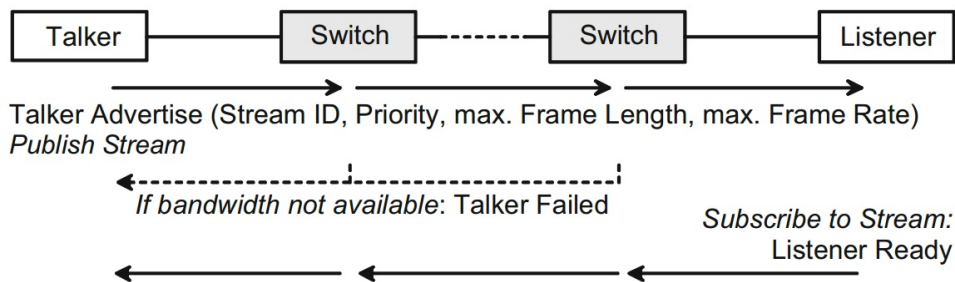


Abbildung 5.10.: Einrichtung einer Echtzeitverbindung mit dem STR[[25]]

In der Abb. 5.10 wird die Einrichtung einer Echtzeitverbindung mit dem STR dargestellt. Ein Sender, der Echtzeitdaten sendet, wird als *Talker* bezeichnet und kündigt seinen Bandbreitenbedarf mit Hilfe einer *Talker Advertise* Botschaft an. Anschließend prüft der verbundene Switch, ob der nötige Pufferspeicher (*Queue*) verfügbar ist und leitet dann die Botschaft an alle anderen Geräte weiter. Der Switch ergänzt die Botschaft um einen Schätzwert, über welchen der Empfänger eine Information über die gesamte Übertragungslatenz der

## 5.5. Echtzeitanwendungen mit Automotive-Ethernet

späteren Echtzeitdaten erhält. Im Falle, dass die nötige Bandbreite nicht verfügbar ist, wird vom Switch eine *Talker Failed* Botschaft ausgesendet.

Ein Steuergerät das Echtzeitdaten empfängt, wird als *Listener* bezeichnet und gibt das mit einer *Listener Ready* Botschaft beim *Talker* an. In dem Moment, wenn eine *Talker Advertise* Ankündigung durch mindestens eine *Listener Ready* Anmeldung bestätigt wurde, wird von allen beteiligten Geräten die nötige Bandbreite reserviert und danach kann die Übertragung beginnen. Echtzeitbotschaften, ebenso wie die Botschaften des MSRP-Protokolls, werden als normale Ethernet-Botschaften (siehe Abb. 2.10) verschickt[[25]].

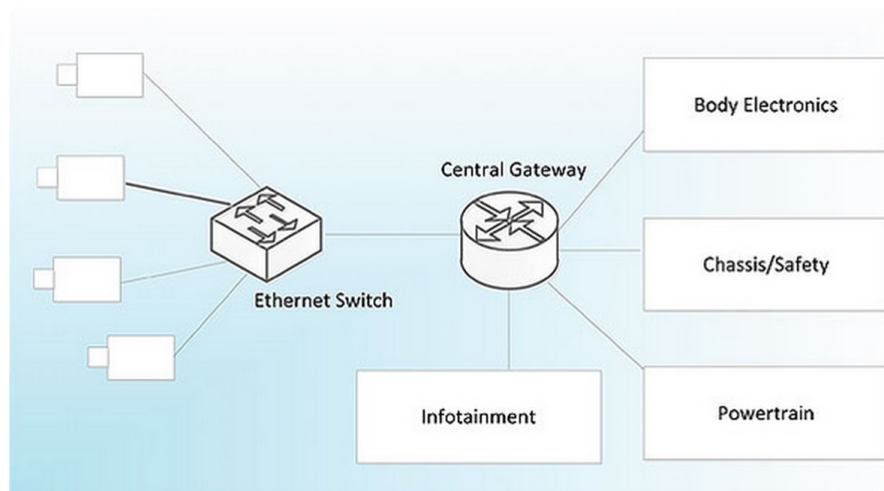


Abbildung 5.11.: Anwendung der IEEE 802.1 AVB Norm in einem Kamerasystem[[4]]

Die Abb. 5.11 zeigt eine Anwendung von IEEE 802.1 AVB am Beispiel eines Kamerasystems. In diesem System werden vier Kameras über den Ethernet-Switch an das zentrale Gateway verbunden. Die Realisierung der Verbindungen zwischen den Geräten als Ethernet-AVB garantiert die Übertragung der Kameradaten in Echtzeit und ermöglicht ebenfalls die Nutzung der Verbindungen für Konfigurations-, Diagnose- und Managementdaten[[4]].

### 5.5.2. Time-Sensitive Networking

Ethernet wurde bereits durch den AVB-Standard um Echtzeitprotokolle für die Übertragung von zeitkritischen Audio- und Videodaten erweitert. Um Ethernet in der Zukunft auch als zentrale Lösung (Backbone) im Fahrzeug einsetzen und die speziellen Anforderungen der Automobilhersteller erfüllen zu können, werden unter der *IEEE 802.1 Time-Sensitive Networking (TSN) Task Group* neue Normen entwickelt. Diese Erweiterungen werden eine beträchtliche Reduktion der Übertragungsdauer bewirken und den Einsatz von Ethernet somit auch für Fahrerassistenz-Systeme und sicherheitskritische Anwendungen ermöglichen. [[4]].



## 5. Ethernet im Kraftfahrzeug

IEEE 802.1 TSN setzt sich aus folgenden Teilnormen zusammen:

- **IEEE 802.1ASbt:** Beschreibt Verbesserungen des Uhrensynchronisations-Protokolls, zur Erhöhung der Widerstandsfähigkeit gegen Störungen.
- **IEEE 802.1Qca:** Definiert den Aufbau von redundanten Datenleitungen, mittels denen Ethernet-Nachrichten gleichzeitig über mehrere unabhängige Pfade versendet werden können.
- **IEEE 802.1Qcc:** Beschreibt Verbesserungen des SRP-Reservierungs-Protokolls.
- **IEEE 802.1CB:** Definiert Mechanismen, mittels denen ein Empfänger Informationen aus redundanten Nachrichten extrahiert.
- **IEEE 802.1Qbv:** *Time-Aware Shaper* für die Realisierung von zeitgesteuerter Kommunikation in Ethernet-Switches[[4]].

Die IEEE-802.1Qbv Mechanismen sind insbesondere für die Übertragung von zeitkritischen Signalen im Fahrzeug wichtig. Diese sind schematisch in der Abb. 5.12 dargestellt.

Betrachtet wird ein physischer Port eines Ethernet-Switches, über welchen mehrere logische Warteschlangen ihre Nachrichten verschicken. Da mehrere Warteschlangen gleichzeitig Nachrichten enthalten können, muss der Ethernet-Switch entscheiden, von welcher Warteschlange er die nächste zu versendende Nachricht entnehmen wird.

Nach IEEE 802.1Qbv wird diese Entscheidung vom Switch, mit Hilfe eines Kommunikations-Schedule (*Gate Event List*) getroffen. Jede dieser Warteschlangen kann durch einen Schalter (*Transmission Gate*) aktiviert oder deaktiviert werden und wird vom Kommunikations-Schedule bestimmt[[4]].

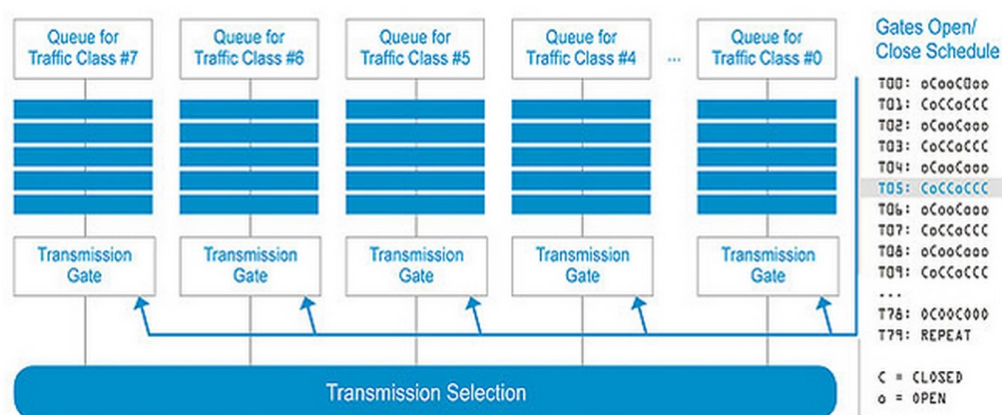


Abbildung 5.12.: IEEE-802.1Qbv-Warteschlangen-Modell mit Kommunikations-Schedules („Gate Event List“)[[4]]



## 5.5. Echtzeitanwendungen mit Automotive-Ethernet

In der Abb. 5.13 wird ein Beispiel eines Kommunikationsszenarios des IEEE 802.1Qbv Entwurfs dargestellt. Das Beispiel zeigt zwei Warteschlangen mit jeweils zeitkritischen und anderen Nachrichten. Der Verlauf der Nachrichtenübertragungen wird am unteren Rand der Abbildung dargestellt.

Wenn die entsprechende Warteschlange aktiv ist, „TA enabled“, werden zeitkritische Nachrichten versendet. Anhand eines Kommunikations-Schedule wird von „TA enabled“ zu „TA disabled“ umgeschaltet. Ein Sender verschickt Nachrichten nur zu Zeitpunkten, zu welchen der Kommunikations-Schedule in den Switches sicherstellt, dass die Warteschlangen mit zeitkritischen Nachrichten abgearbeitet wurden[[4]].

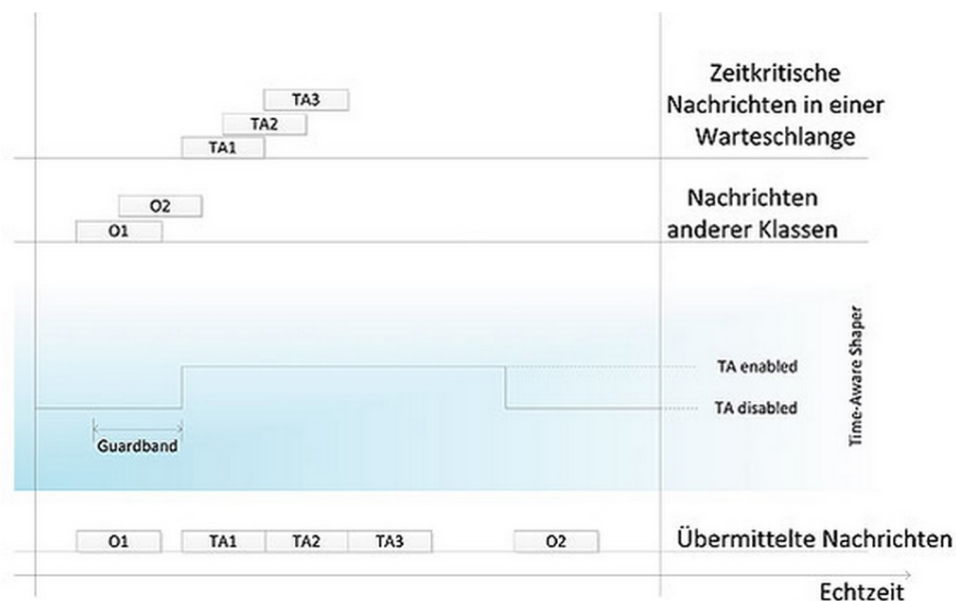


Abbildung 5.13.: IEEE 802.1Qbv Kommunikations-Szenario[[4]]

IEEE 802.1Qbv standardisiert eine Form von zeitgesteuerter Kommunikation, die ähnlich der von FlexRay ist. Ein wesentlicher Unterschied zu FlexRay ist, dass in IEEE 802.1Qbv ein Zeitpunkt der Zeitsteuerung typischerweise mehrere Ethernet-Nachrichten betrifft. Im Gegensatz betrifft ein Zeitpunkt der Zeitsteuerung in FlexRay und ähnlichen zeitgesteuerten Protokollen immer nur eine Nachricht. TSN bietet im Vergleich zur AVB Vorteile bei der Latenzzeit und im Jitter-Verhalten.

Trotz dessen, dass der Standardisierungsprozess bei TSN noch nicht abgeschlossen ist, sind bereits konkrete Produkte erhältlich (z.B. von TTTech). Diese unterstützen sowohl IEEE AVB als auch den Draft-Standard IEEE 802.1Qbv und stellen zusätzlich noch spezifische Erweiterungen in Richtung Fehlertoleranz und Netzwerk-Partitionierung bereit[[4]].

## 5. Ethernet im Kraftfahrzeug

### 5.5.3. Interspersed Express Traffic

Um eine schnelle Reaktionszeit bei diversen Funktionen im Fahrzeug gewährleisten zu können, benötigen verschiedene Steuergeräte eine Kommunikationslatenz im Mikrosekundenbereich. Bei Standard-Ethernet wartet ein neues Paket so lange, bis der Sendevorgang des vorhandenen Pakets beendet ist. Das kann bei Gigabit-Geschwindigkeiten sogar hunderte Mikrosekunden dauern, weshalb es die Anforderungen im Fahrzeug nicht erfüllt[[17]].

Diese Problematik versucht man in der IEEE 802.3br *Interspersed Express Traffic* (IET)<sup>3</sup> Gruppe zu lösen. Darin wird ein Standard für industrielle und automotiv Anwendungen entwickelt, welcher die maximale Latenz innerhalb eines Knotens so weit reduziert, dass sie nahe der Zeit für ein kleinstmögliches Paket (64 byte) inklusive IPG (*InterPacket Gap*) liegt.

Pakete hoher Priorität (*Express packets*) können normale Pakete unterbrechen, welche erst nach deren Versenden an die Reihe kommen. Ein unterbrechbares Paket kann mehrere Male durch Express-Pakete unterbrochen werden. Diese Express-Pakete können Latenzen im Mikrosekunden-Bereich gewährleisten[[17]]. Die Fertigstellung des Standards ist für Dezember 2015 vorgesehen.

---

<sup>3</sup><http://www.ieee802.org/3/br/index.html/>, Zugriffsdatum: 11.03.2018

## 5.6. Enabling Technologies

### 5.6.1. DoIP

Durch die immer komplexeren Infotainment- und Fahrerassistenzsysteme steigt die Softwaregröße in Fahrzeugen und damit auch die benötigte Zeit bei der Flash Programmierung (End of Line Programmierung). Durch Verwendung der üblichen CAN-Diagnoseschnittstelle (max. 30 KByte/s) dauert eine Software-Aktualisierung von 100 MByte fast eine Stunde. Deswegen verwenden Fahrzeuge der Oberklasse heutzutage eine zusätzliche Ethernet-Schnittstelle und führen die Übertragung mit dem Standard-Internetprotokoll TCP/IP durch (Abb. 5.14)[[25]].

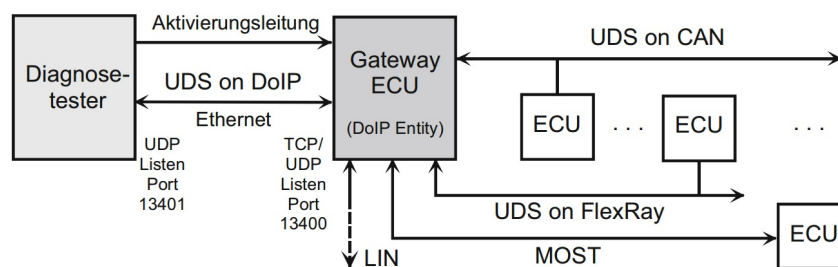


Abbildung 5.14.: Ethernet-Schnittstelle zum Fahrzeug[[25]]

Das Transportprotokoll *Diagnostics over Internet Protocol* (DoIP) wurde entwickelt um vorhandene Diagnoseprotokolle wie *Unified Diagnostic Services* (UDS) sowohl auf der Steuergeräteseite als auch im Diagnosetester weiter nutzen zu können und die Anfragen des Diagnosetesters vom Gateway-Steuergerät direkt zu den internen CAN, FlexRay und LIN Bussystemen durchleiten zu können.

DoIP ermöglicht gewöhnliche UDS-Nachrichten in TCP/IP-Botschaften (UDP oder TCP) zu verpacken und diese über Ethernet oder WLAN zu übertragen (Abb. 5.15). Durch diese standardisierten Protokollschichten soll sowohl die Kompatibilität zwischen Fahrzeugen als auch zwischen einem Fahrzeug und einem Diagnose-Computer gegeben sein.[[25]].

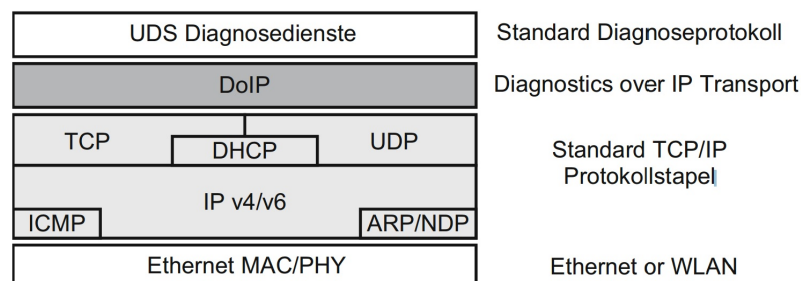


Abbildung 5.15.: DoIP-Transportprotokoll im OSI-Schichtenmodell[[25]]

## 5. Ethernet im Kraftfahrzeug

Da die Anforderungen in der Werkstatt- und Fertigungsumgebung, im Gegensatz zur *Onboard*-Umgebung, unproblematisch sind, wird bei der DoIP-Diagnoseschnittstelle die Standard-Ethernet Technologie eingesetzt. Die 10/100 Mbit/s Ethernet-Verbindung nutzt freie Anschlüsse des *On Board Diagnose* (OBD)-Diagnosesteckers. In der nächsten Zeit soll eine für höhere Übertragungsraten (Gbit/s) geeignete Steckverbindung eingeführt werden. DoIP ist als ISO 13400 standardisiert[[25]].

### 5.6.2. PoE

Über Power over Ethernet (PoE) können Geräte wie z.B. Kameras mit Strom versorgt werden und so die Verkabelungskosten und das Gewicht im Fahrzeug weiter reduziert werden. Die IEEE 801.3bu *1-Pair Power over Data Lines* (PoDL) *Task Force*<sup>4</sup> arbeitet an der Standardisierung von PoE über ein einziges Leitungspaar (Twisted Pair). Die Genehmigung des Standards wird in 2015 erwartet[[17]].

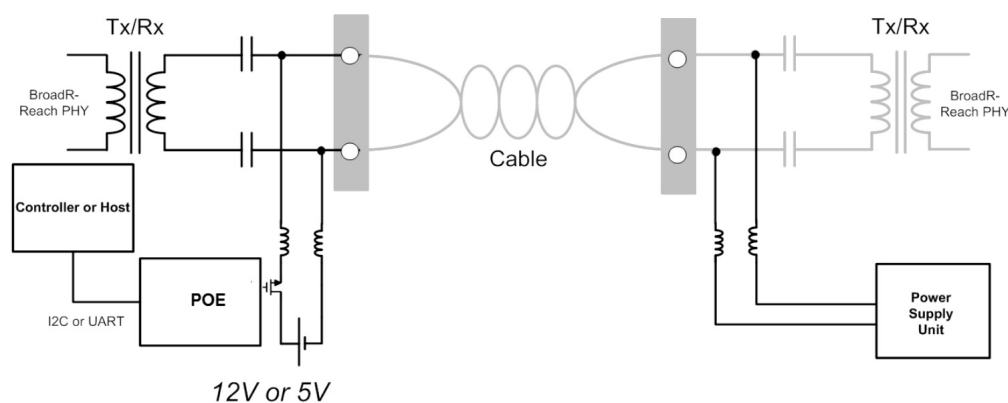


Abbildung 5.16.: Schaltplan der Power over BroadR-Reach® Ethernet Technologie[[21]]

Die Abb. 5.16 zeigt den Schaltplan der *Power over BroadR-Reach® Ethernet* Technologie. PoE ermöglicht deutliche Einsparungen bei Leitungen und Steckern, da Strom und Daten über ein einziges Leitungspaar fließen. Weniger Kabel brauchen weniger Platz, was wiederum zusätzliche Flexibilität in der Anwendung bei Fällen mit knappen Platzverhältnissen bietet. Weniger Stecker bedeutet höhere Zuverlässigkeit, da eine geringere Chance für Fehler oder Korrosion besteht[[21]].

<sup>4</sup><http://www.ieee802.org/3/bu/index.html/>, Zugriffsdatum: 11.03.2018

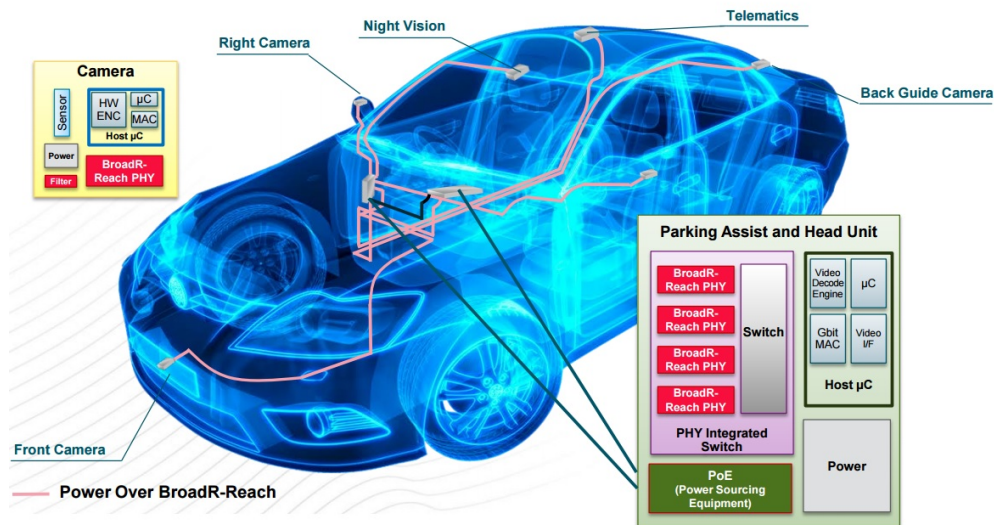


Abbildung 5.17.: Anwendungsfall: Power over BroadR-Reach® Ethernet[[21]]

In der Abb. 5.17 wird ein Anwendungsfall mit *Power over BroadR-Reach® Ethernet* dargestellt. Endgeräte wie Kameras werden über PoE mit Strom versorgt. PoE bietet die Möglichkeit Kameras und andere Geräte am Netzwerk selektiv ein- und auszuschalten. Weiterhin ermöglicht es die Überwachung der Leistungsaufnahme von jedem Gerät im Netzwerk und einen Schutz vor Überlastung[[21]].

### 5.6.3. Energy-Efficient Ethernet

Nicht alle elektrischen Komponenten im Fahrzeug schalten sich aus wenn der Motor abgestellt wird. Um den Stromverbrauch bei ein- und ausgeschalteten Motor zu minimieren, wird *Energy-Efficient Ethernet (EEE)* verwendet. EEE schaltet das Netzwerk aus, wenn es nicht in Gebrauch ist. Der EEE Standard wird von der P802.3az *Energy Efficient Ethernet Task Force*<sup>5</sup> entwickelt.

Durch das *Energy Detect Module (EDM)* (Abb. 5.18) welches die Link-Aktivität detektiert können einzelne Komponenten bei Bedarf eingeschaltet werden. Netzwerksegmente von Komponenten, welche im ausgeschalteten Zustand (Motor aus) niemals eingeschaltet sein müssen, werden komplett ausgeschaltet. Diejenigen, die eingeschaltet bleiben müssen, werden EEE benutzen, um den Stromverbrauch zu minimieren[[17]].

<sup>5</sup><http://www.ieee802.org/3/az/>, Zugriffsdatum: 11.03.2018

## 5. Ethernet im Kraftfahrzeug

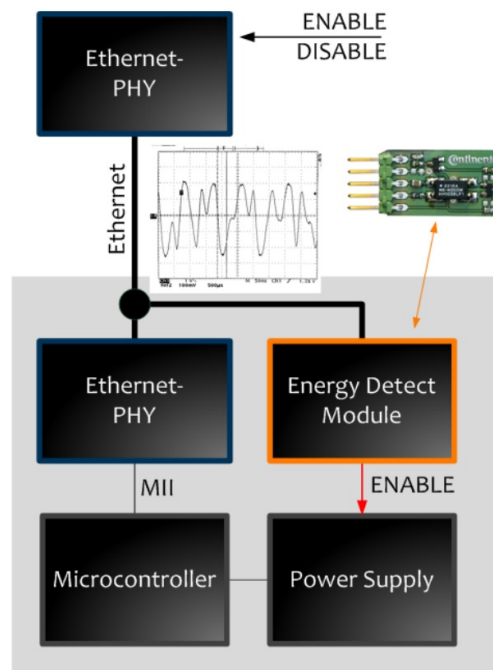


Abbildung 5.18.: EEE: Selektiver *wake up* Mechanismus durch EDM (Quelle: Continental)

### 5.7. Teststandards für Automotive-Ethernet

Um die Fehlerfreie Funktionalität aller Komponenten im Automotive-Ethernet-Netzwerk gewährleisten zu können müssen laufend neue Teststandards entwickelt werden. In der folgenden Aufzählung werden verschiedene Teststandards beschrieben:

- **Konformitäts-Tests:** Entwickler und Hersteller sind auf die Konformitäts-Tests angewiesen, um die Einhaltung von Protokollen und der Interoperabilität zu validieren. Widerstandsfähigkeit gegenüber unerwarteten Protokollverhalten ist auch ein kritischer Testpunkt. Die folgenden Konformitäts-Tests sind notwendig und werden bereits für Automobiltests verwendet: TCP/UDP/IP Tests für Automotive-Stacks und Komponenten, AUTOSAR-, AVB-, IET- und IEEE-Konformitäts-Tests[[17]].
- **Protokoll-Validierung und Performance-Tests:** Es besteht ein eindeutiger Bedarf nach Performance-Tests auf der Applikations- und Protokollebene. Solche Performance-Tests garantieren, dass, wenn Applikationen unter realen Bedingungen eingesetzt werden, diese eine qualitativ hohe Benutzererfahrung bieten. Solche Tests umfassen: Performance der Automobil-Anwendungen, 802.1AS Zeitsynchronisation (Genauigkeit und Performance), A/V Qualität, IET Performance und AUTOSAR Performance.

- **PHY-Tests:** PHY-Hersteller bitten um Hilfe beim Testen von 100 Mbit/s (BroadReach®) und 1GE (802.3bp 1000Base-T1) PHYs. Physikalische Tests könnten eine Vielzahl von wichtigen Informationen liefern, wie zum Beispiel die Sicherstellung von Anforderungen einer Spezifikation, die Demonstration des *Proof of Concept*, die Bereitstellung von Testdaten für die Weiterentwicklung und Qualitätsbestätigung sowie die Bestätigung der Eignung für Endverwendung.
- **Stress- und Zuverlässigkeitstests von Komponenten:** Die Zuverlässigkeit von Automobilkomponenten ist sicherheitskritisch. Jede Komponente muss unter allen möglichen Betriebsbedingungen getestet werden. Dies erfordert oft Tests bei extremen Temperaturen/Bedingungen, bei welchen das Gerät über einen längeren Zeitraum läuft.
- **Switch-Tests:** Automotive Ethernet-Switches werden die Unterstützung für die neuen AVB und IET (Institution of Engineering and Technology) Funktionen testen müssen. Folgende Punkte müssen getestet werden: Protokolle für Warteschlangen, Verhalten beim Paketverlust, Bandbreite und Latenz, Verhalten wenn ein mit dem Switch verbundenes Gerät Fehlverhalten aufweist, Test für Ausfallsicherung und Konvergenzzeiten, Negative Testfälle, 2544/2889 Tests mit den oben genannten Bedingungen[[17]].
- **Produktions-Tests:** Der Kabelbaum, alle Komponenten und das komplette Fahrzeug müssen validiert werden. Die Tests müssen bestätigen, dass Verbindungen korrekt sind und die Kommunikation zuverlässig funktioniert (richtige Impedanz und Geschwindigkeit, kein Datenverlust), CPUs korrekt arbeiten und dass alle angeführten Aufgaben in verschiedenen Bedingungen problemlos arbeiten.
- **Sicherheits-Tests:** Fahrzeuge werden immer mehr mit der Umgebung und dem Internet verbunden. Das erhöht die Gefahr des externen *Hackings* (Eindringens), für welches schon einige Handbücher (für einige Automodelle) verfügbar sind. Deshalb wird Netzwerksicherheit immer wichtiger in Fahrzeugen. Die Automobilhersteller müssen sicherstellen, dass Fahrzeuge mehrere fehlersichere Systeme besitzen, um das Risiko des Eindringens zu minimieren, vor allem in der Antriebstrang-Domäne.
- **Fahrzeug-Diagnose:** Die Fahrzeug-Diagnose kann intern in einem Board-Computer eingebaut sein oder sie befindet sich extern in einer Werkstatt. Die externe Diagnose kann das gesamte Fahrzeug oder nur einzelne Komponenten überprüfen. Solche Diagnostiktests müssen in der Lage sein, die Diagnoseausrüstung zu simulieren und mit allen Fahrzeug-Domänen fehlerfrei kommunizieren zu können[[17]].



## 5. Ethernet im Kraftfahrzeug

### 5.8. Ausblick

Zurzeit befindet sich Automotive-Ethernet am Anfang der 2. Generation (Abb. 5.19). Das heißt, es sind heuer die ersten Serienfahrzeuge in Produktion gegangen, welche die BroadR-Reach® Ethernet Technologie verwenden. Diese soll zuerst im Infotainment-Bereich und für Fahrerassistenzsysteme (ADAS) eingesetzt werden.

Da die Anforderungen an die Übertragungsraten immer höher werden (100 Mbit/s ist für Videoübertragung ausreichend, aber zu wenig um als Backbone im Fahrzeug zu agieren), wurde die Entwicklung des Nachfolgers von *100 Mbit/s Twisted Pair Standards* in die Wege geleitet. Dafür wurde die *IEEE P802.3bp 1000BASE-T1 PHY Task Force*<sup>6</sup> gegründet, welche einen Standard für Übertragungsraten bis 1 Gbit/s über eine verdrehte Zwei-Draht-Leitung entwickeln soll.

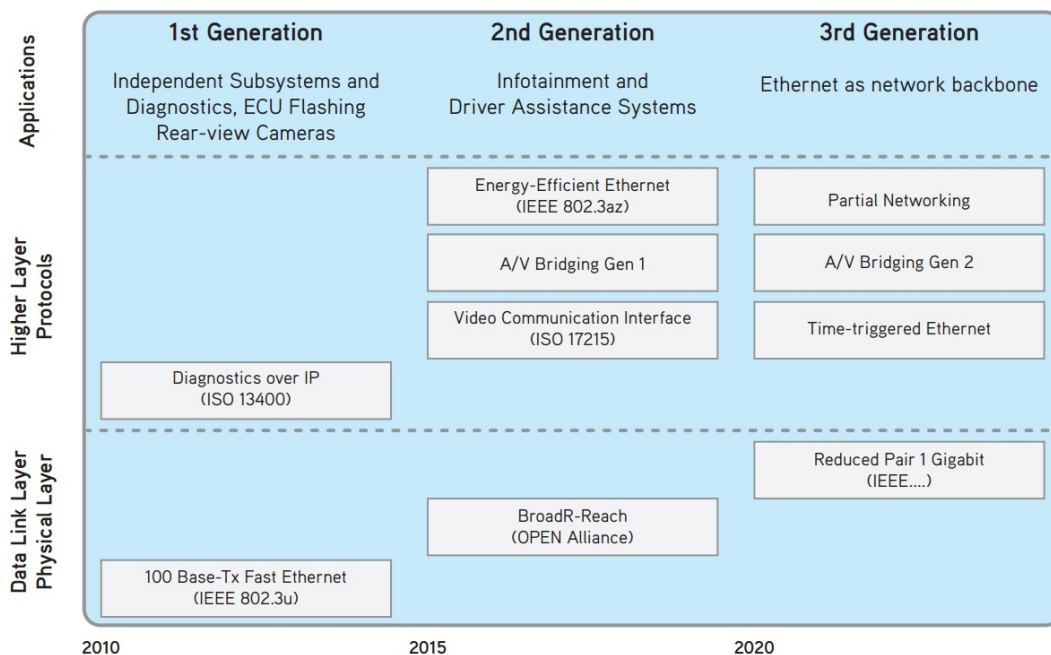


Abbildung 5.19.: Entwicklung vom Automotive-Ethernet bis 2020[[17]]

Basierend auf den aktuellen Schätzungen wird erwartet, dass Prototypen in 2015/2016 erhältlich sein werden und 1 Gbit/s Ethernet bis 2017 Standard wird. Im Consumer-Bereich (Standard-Ethernet) bewegt man sich zurzeit in Richtung 40 Gbit/s bis 100 Gbit/s, weshalb es viel Raum für Weiterentwicklung gibt.

<sup>6</sup><http://www.ieee802.org/3/bp/>, Zugriffsdatum: 11.03.2018



### 5.8.1. Netzwerkarchitektur mit Automotive-Ethernet

Bis zum Jahr 2017 (Abb. 5.20) werden nach Schätzungen alle Fahrerassistenzsysteme (ADAS) und der Infotainment-Bereich Ethernet verwenden. Das heißt dass Ethernet voraussichtlich MOST und LVDS aus den meisten Fahrzeugen verdrängen wird.

Die sicherheitskritische Kommunikation im Antriebsstrang und Safety-Bereich wird weiterhin von CAN/CAN-FD und FlexRay übernommen. Die Anforderungen an die Bandbreite für ADAS und das Infotainment-System steigen immer weiter.

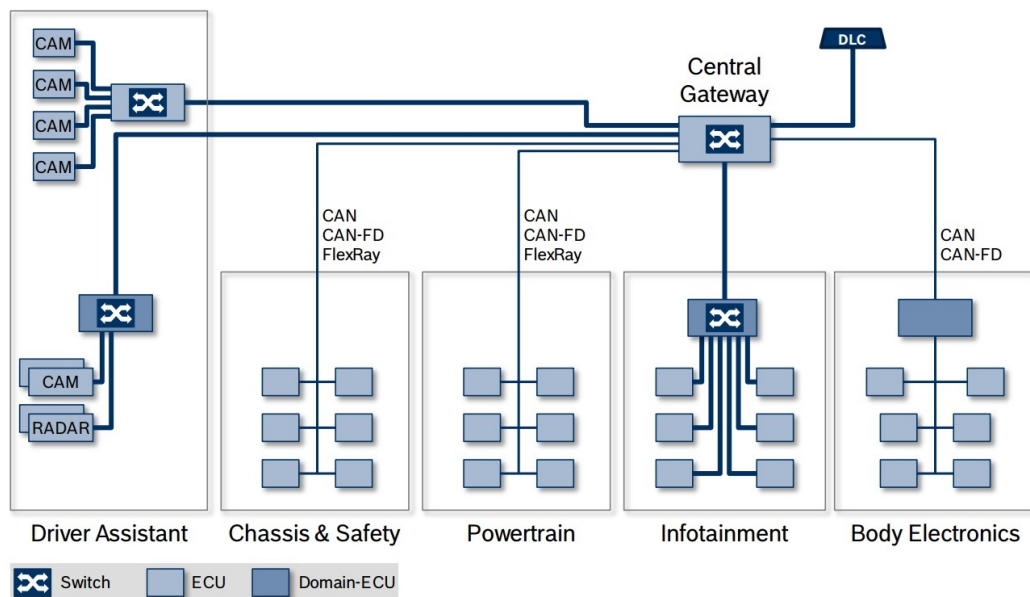


Abbildung 5.20.: Automotive Netzwerkarchitektur 2015-2017[[12]]

Es wird geschätzt dass bis zum Jahr 2020 die Kosten für Elektronik im Fahrzeug 40% betragen werden (heute 32%). Diese werden von neuen Kommunikations-, Infotainment- und Sicherheitsfunktionen vorangetrieben.

Der Fahrzeug-Kabelbaum wird sich aus heterogenen Netzwerken, bestehend aus proprietären Protokollen wie MOST und CAN, in hierarchische homogene Ethernet-Netzwerke umwandeln (Abb. 5.21).

## 5. Ethernet im Kraftfahrzeug

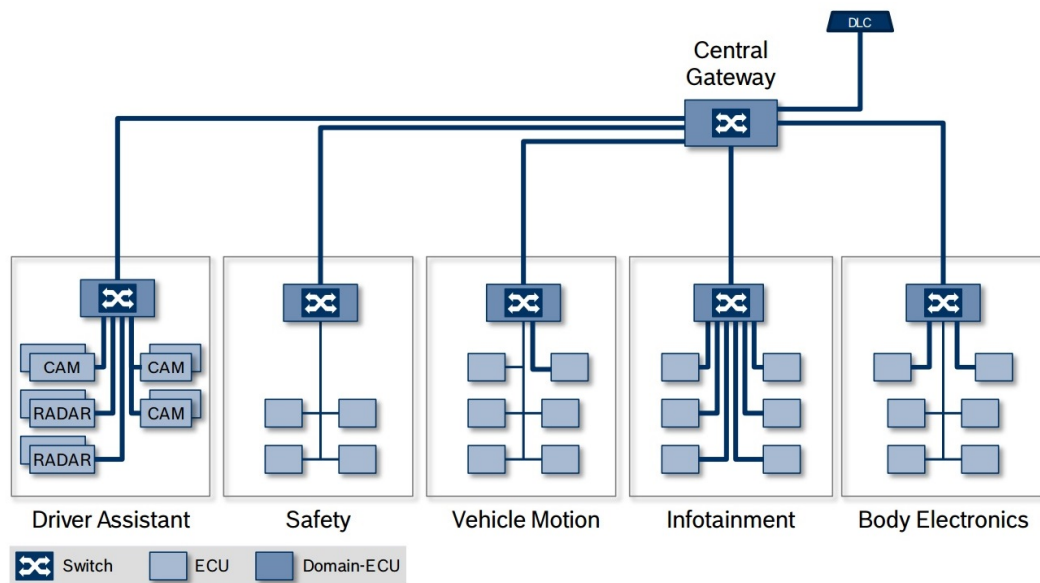


Abbildung 5.21.: Voraussichtliche Automotive Netzwerkarchitektur in 2020[[12]]

In der neuen Netzwerkarchitektur wird *switched 1GE Automotive Ethernet* alle Domänen im Fahrzeug miteinander verbinden. Dies bedeutet, dass Ethernet ein Shared Medium werden wird, wo Signale für die Gaspedalsteuerung (sicherheitskritisch) über das gleiche Leitungspaar (Twisted Pair) versendet werden, wie das Signal für Wechseln von einem Radiosender[[17]].

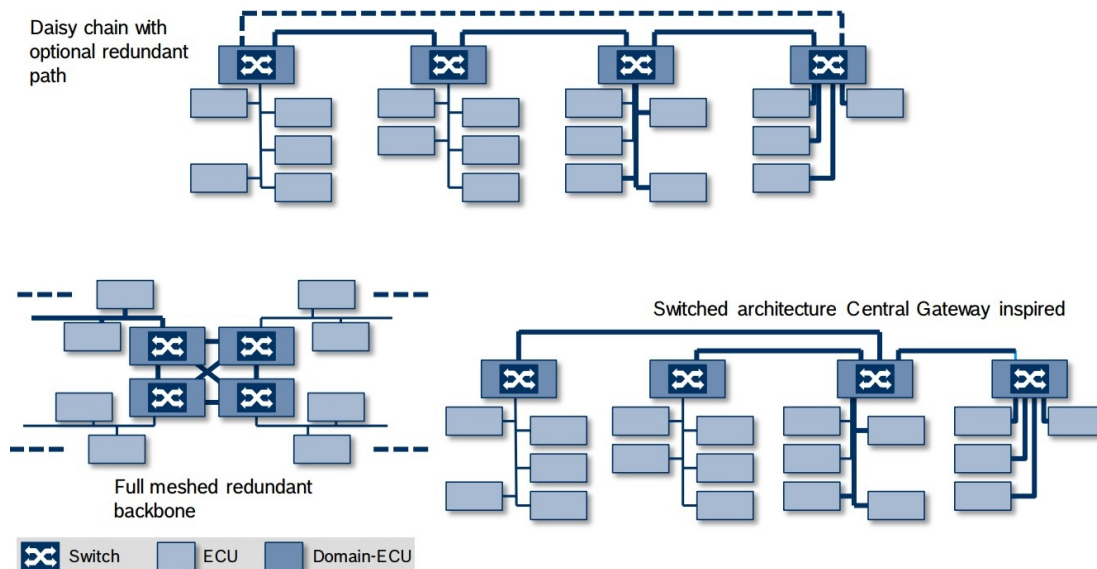


Abbildung 5.22.: Verschiedene Backbone-Architekturen mit Automotive Ethernet[[12]]

Die Abb. 5.22 zeigt verschiedene Implementierungen der Backbone-Architektur mit Automotive-Ethernet. Diese neue Fahrzeug-Netzwerkarchitektur hilft die Kosten und das Gewicht zu verringern und vereinfacht die Kommunikation zwischen den verschiedenen Systemen in- und außerhalb des Fahrzeugs[[17]].

### 5.8.2. Marktpotenzial

Im Jahr 2014 wurden weltweit ungefähr 94 Millionen Autos verkauft. Diese Zahl wächst um rund 3,3% jedes Jahr (in den letzten 9 Jahren). Es wird geschätzt, dass bis zum Jahr 2020 über 117 Millionen Neuwagen pro Jahr verkauft werden. Die Abb. 5.23 zeigt die Anzahl der verkauften Neuwagen weltweit.

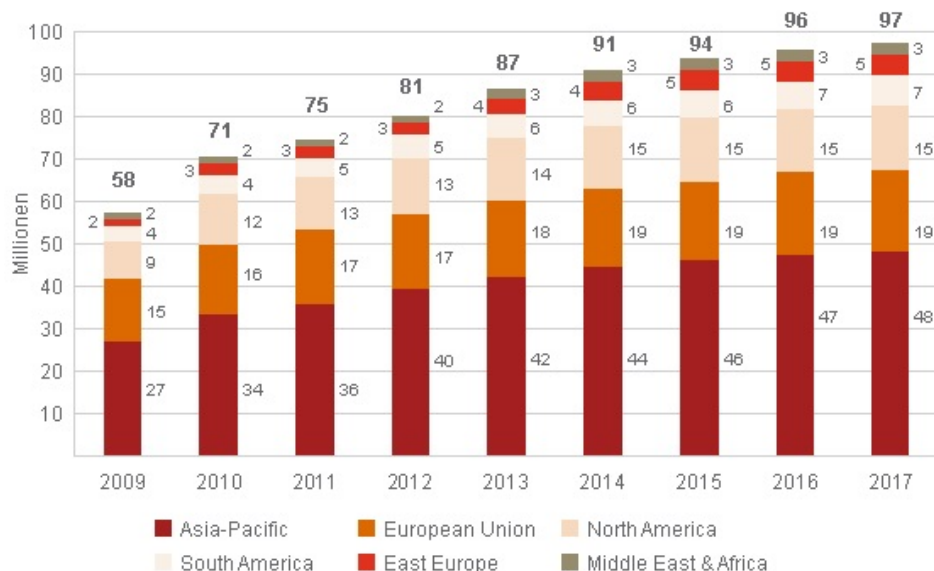


Abbildung 5.23.: Anzahl der weltweit verkauften Neuwagen über die vergangenen Jahre (Quelle: PwC)

Der Verkauf von Luxus-Autos stieg in den letzten Jahren und beträgt über 5% der neu verkauften Fahrzeugen. Ein Luxuswagen besitzt fünf mal so viel elektronische Komponenten wie ein Mittelklassewagen.

In 2012 wurden weltweit fast 1.6 Millionen Hybrid- und Elektrofahrzeuge verkauft. Dieser Markt wächst über 200% jedes Jahr. Im Gegensatz zu einem rein Kraftstoff betriebenen Fahrzeug haben Hybrid- und Elektrofahrzeuge drei mal so viel elektronische Komponenten[[17]].

Wie man aus der Abb. 5.24 sehen kann halten sich die Kosten pro Ethernet-Knoten ungefähr im Bereich von FlexRay und MOST. Da die Preise pro Ethernet-Knoten in Zukunft immer

## 5. Ethernet im Kraftfahrzeug

weiter fallen werden, während gleichzeitig die Bandbreite steigen wird, wird es für die konkurrierenden Technologien schwer mit Ethernet mitzuhalten.

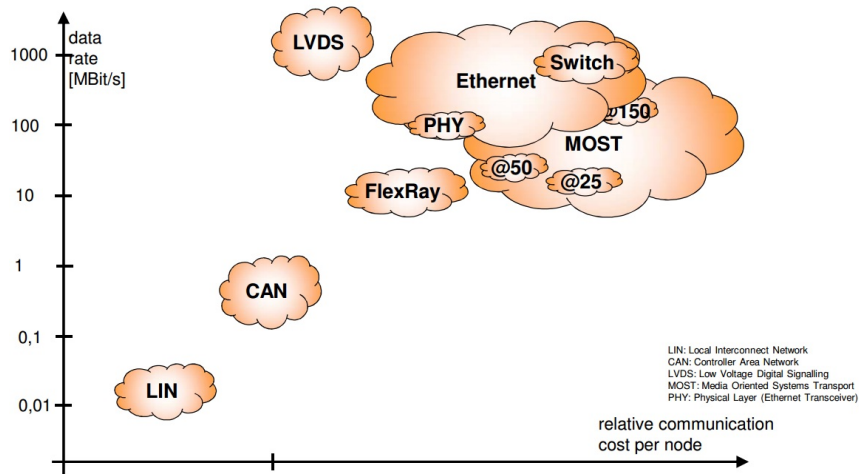


Abbildung 5.24.: Vergleich der relativen Kommunikationskosten pro Knoten im Fahrzeug[[19]]

Bis zum Jahr 2019 werden Mittelklasse-Fahrzeuge ungefähr 8 bis 20 Ethernet-Ports besitzen. In Luxusfahrzeugen soll diese Zahl bis 50 steigen und Hybrid- und Elektrofahrzeuge sollen fast 80 Ethernet-Ports enthalten. In der Abb. 5.25 wird die voraussichtliche Anzahl von Ethernet-Ports im Fahrzeug dargestellt[[12]].

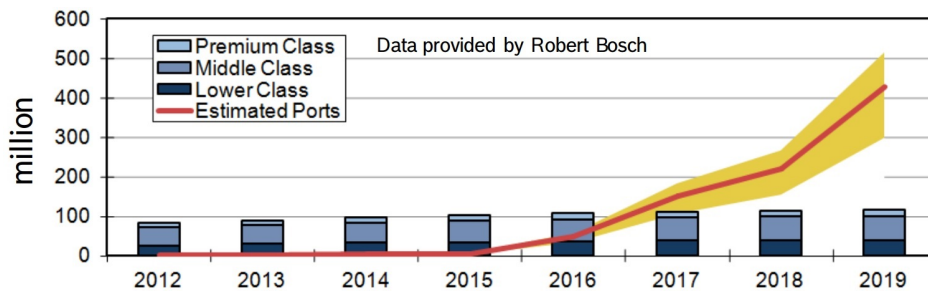


Abbildung 5.25.: Voraussichtliche Anzahl von Ethernet-Ports im Fahrzeug bis 2019[[12]]

Nach einer Studie von Frost & Sullivan werden bis 2020 40% der Neuwagen Ethernet verwenden. Sie schätzen weiter, dass 300 Millionen Automotive-Ethernet-Ports im Jahr 2020 verkauft werden[[17]].

Im Anschluss befindet sich eine kurze Zusammenfassung auf das Thema „Ethernet im Kraftfahrzeug“ (Factsheet - Ethernet im Kraftfahrzeug).

### Motivation für Ethernet

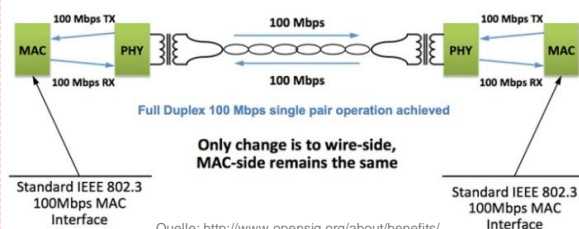
- Zunehmende **Vernetzung** des Fahrzeugs intern und mit der Fahrzeugumgebung (Car2x)
- Einzug der „**Consumer Electronic**“ ins Fahrzeug (einfache Plug & Play Anbindung von Endgeräten und Apps)
- Hohe und günstige **Übertragungs-Bandbreiten**
- **Bewährte Technologie**, weit verbreitet
- Viele Entwicklungs- und Diagnosewerkzeuge bereits vorhanden und erprobt

### Vorteile zu anderen Bussystemen

- **Hohe Bandbreiten** – bei BroadR-Reach® Ethernet stehen 100Mbit/s per Verbindung und Richtung zur Verfügung
- Große Anzahl an verwendbaren **Protokollen, Technologien und Applikationen aus der PC- und Industrie-Ethernet-Welt** vorhanden
- Einfache **Skalierbarkeit** des Netzwerks durch Switches
- **Kostenvorteil** (Verkabelung und Bausteine) sowie **Gewichtsreduzierung** (Verkabelung)
- **Großes Entwicklungspotenzial** - Zurzeit 100Mbit/s im Fahrzeug, währenddessen geht es im Consumer Bereich in Richtung 40Gbit/s bis 100Gbit/s (viel Raum für Weiterentwicklung)

### Unterschiede zum Standard Ethernet

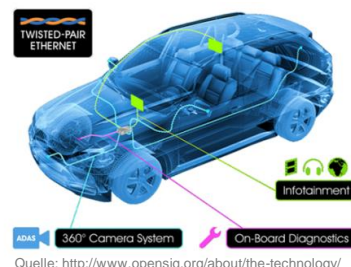
- In der **Automobil Umgebung** gelten andere Anforderungen als für Computer Netzwerke in Gebäuden (Störsicherheit, Gewicht, Umwelteinflüsse, Echtzeitfähigkeit, Kosten, Länge der Leitungen usw.)
- Um Ethernet Automobil tauglich zu machen, wurde die **BroadR-Reach®** Technologie entwickelt, welche den Standard-Ethernet-Hardware Abschnitt ersetzt und so eine 100Mbit/s Kommunikation im Full-Duplex Modus über kostengünstige ungeschirmte Single-Twisted-Pair-Kabel ermöglicht



- Die **OPEN Alliance SIG Group** wurde gegründet um BroadR-Reach® als offenen Standard für automobiles Ethernet zu etablieren

### Herausforderungen

- Störsicherheit und Umweltbeständigkeit sind noch in der **Praxis zu bestätigen**
- Architektur der vorhandenen Steuergeräte ist auf Ethernet auszulegen (**erhöhte Hardware- und Softwareanforderungen**)
- Anpassung der **Test- & Entwicklungswerkzeuge** für automobiles Ethernet
- **Vollständige Integration in AUTOSAR**
- **Neue Netzwerkarchitekturen** (Switched Network) erforderlich
- Anzahl der **Lieferanten für Automotive-Ethernet Bausteine** ausweiten (derzeit Broadcom, Freescale & NXP)



### Anwendungsfälle

- **Diagnosezugang** – Leistungsfähigere Diagnose (z.B. Real-Time Monitoring) und schnelleres Flashen von Steuergeräten
- **Infotainment Bereich** – „Digital-Home“ im Fahrzeug. Nutzung der selben Internet und Multimedia Dienste (z.B. Audio-Video-Streaming) wie in der Wohnung
- **ADAS** (fortschrittliche Fahrerassistenzsysteme) – Bilder und Videostreams der Fahrzeug-Kameras (IP-Kameras), Car2x-Kommunikation

### Ausblick

- **Serieneinsatz** – in den nächsten Jahren werden die meisten Designs für Kameranetzwerke, Infotainment und ähnliche auf Ethernet basieren
- Übertragungsraten im **Gbit/s** Bereich (IEEE 802.3 Reduced Twisted Pair Gigabit Ethernet PHY study group)
- Mithilfe von Switches schafft Ethernet eine **hierarchische Netzwerkarchitektur**
- Ethernet wird das primäre Bord-Netzwerk für Fahrzeuge (**Ethernet als Backbone**)
- Einsatz von **Ethernet für sicherheitskritische Anwendungen** (TTEthernet)



## 6. Praktische Arbeit

*In diesem Kapitel wird die Entwicklung und Inbetriebnahme des Standard- und Automotive-Ethernet-Demonstrators beschrieben. Eine ausführlichere Beschreibung der Inbetriebnahme und der verwendeten Programme befindet sich im Dokument "Dokumentation - CAN to Ethernet Verbindung".*

### 6.1. Ziele

Im Praktischen Teil der Masterarbeit sollten auf Basis zweier Evaluierungsboards (Abb. 6.2) von Texas Instruments (TI) zwei *CAN to Ethernet* Demonstratoren aufgebaut werden.

Im Standard-Ethernet-Demonstrator (Abschn. 6.2) sollte eine Schnittstelle zwischen CAN und Ethernet hergestellt werden. Das heißt, die empfangenen CAN-Nachrichten sollen auf ein Ethernet-taugliches Protokoll umgesetzt (UDP oder TCP) und über die Ethernet-Schnittstelle zum anderen Evaluierungsboard verschickt werden. Diese Ethernet-Nachricht soll daraufhin entpackt und als CAN-Nachricht aus dem CAN-Modul wieder versendet werden.

Dieser Datenfluss soll in beide Richtungen gleichzeitig stattfinden, also im Vollduplex-Modus.

Im Automotive-Ethernet-Demonstrator (Abschn. 6.3) sollten die Standard-Ethernet-Transceiver durch die auf der BroadR-Reach-Technologie (siehe Abschn. 5.4.1) basierenden automobiltauglichen Transceiver von Broadcom ersetzt werden. Dafür sollten zwei Platinen entworfen werden, welche die Evaluierungsboards um eine BroadR-Reach-Ethernet-Schnittstelle erweitern. Der Datenfluss ist identisch dem Standard-Ethernet-Demonstrator. Anschließend sollten Untersuchungen hinsichtlich automobiler Tauglichkeit der Ethernet-Technologie bewertet werden.

## 6. Praktische Arbeit

### 6.2. Demonstrator - Standard Ethernet

Beim Standard-Ethernet-Demonstrator sollte anhand zweier Evaluierungsboards (Abb. 6.2) von *Texas Instruments* (TI) eine Schnittstelle zwischen CAN und Ethernet realisiert werden.

Nachrichten werden in die CAN-Schnittstelle reingeschickt. Daraufhin wird das Datensegment der CAN-Nachricht auf ein Ethernet-taugliches Protokoll umgesetzt (UDP oder TCP) und über den Ethernet-Transceiver zum anderen Evaluierungsboard geschickt. Das Datensegment (Payload) der empfangenen Ethernet-Nachricht wird in eine CAN-Nachricht verpackt und aus der CAN-Schnittstelle rausgeschickt (siehe Abb. 6.1).

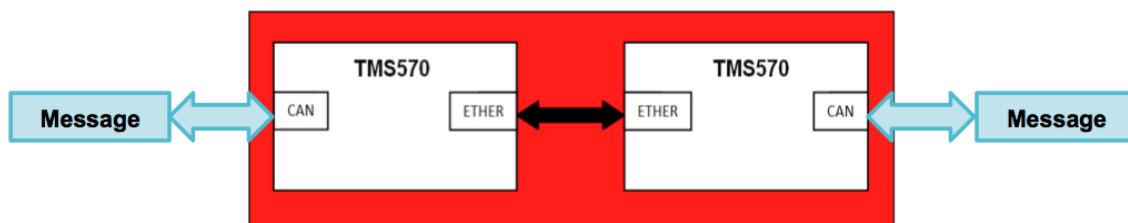


Abbildung 6.1.: Datenfluss des Standard-Ethernet-Demonstrators

An den CAN-Analysern, welche mit den CAN-Modulen der Evaluierungsboards verbunden werden, kann man die gesendeten und empfangenen Nachrichten betrachten. Dieser Datenfluss findet in beiden Richtungen gleichzeitig statt, also im Vollduplex-Modus.

Tabelle 6.1.: Konnektoren am TMS570-Evaluierungsboard

Konnektor	J1	J2	J3	J4	J5	J6	J9	J10	J11
Funktion	Ethernet RJ45	CAN1	CAN2	2,5mm +12V in	Mini USB JTAG	S2 DIP Switch	Steckleiste	Steckleiste	Steckleiste

Die Tabelle 6.1 gibt eine Beschreibung der einzelnen Konnektoren am Evaluierungsboard (Abb. 6.2).



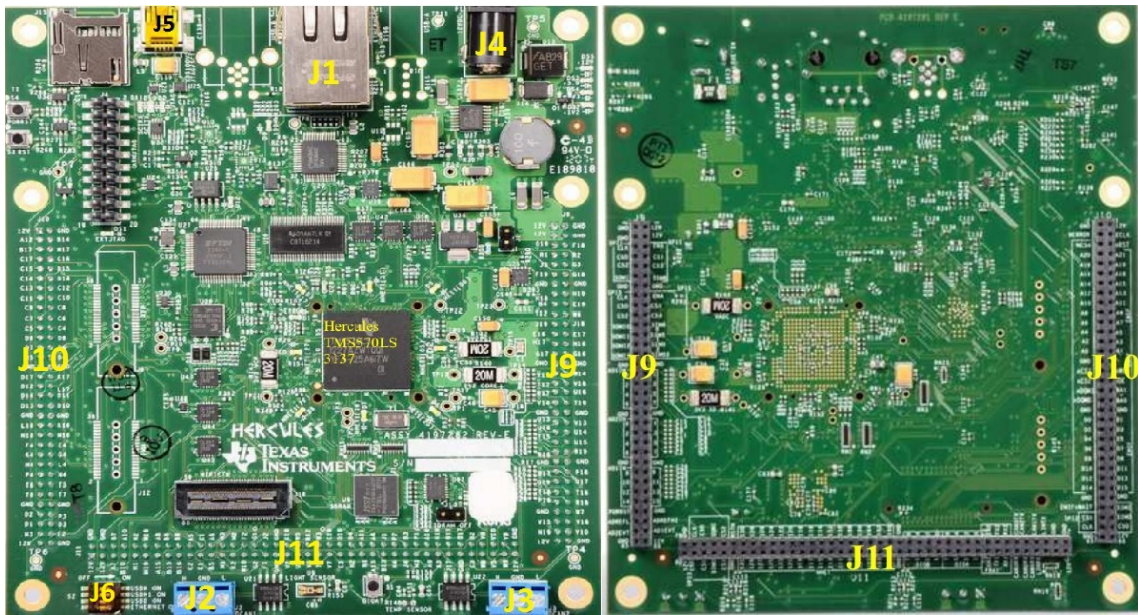


Abbildung 6.2.: Evaluierungsboard von TI - TMS570LS31x Hercules Development Kit[[16]]

### 6.2.1. Verwendete Werkzeuge

Für die Softwareentwicklung wurde die integrierte Entwicklungsumgebung *Code Composer Studio* (CCS)<sup>1</sup> verwendet. CCS ist eine in Eclipse integrierte Entwicklungsumgebung von TI, vorrangig für Embedded-Entwicklung von TI-s Mikrocontrollern und ARM-basierten Prozessoren.

Mit Hercules Peripheral Drivers (HALCoGen)<sup>2</sup> von TI wurden über das GUI-basierte Konfigurations-Tool Peripherietreiber erstellt und konfiguriert.

Für das Versicken und Empfangen der CAN-Nachrichten von und zum Evaluierungsboard wurde der CAN-Analyser *USB-to-CAN Compact*<sup>3</sup> von IXXAT verwendet.

MiniMon ist ein einfaches CAN-Monitoringprogramm von IXXAT, mit welchem die über den CAN-Analyser verschickten CAN-Nachrichten und der dementsprechende Busverkehr auf dem CAN-Bus beobachtet wurde.

Eine detaillierte Beschreibung der Installation und Bedienung der einzelnen Programme befindet sich im separaten Dokument *”Dokumentation - CAN to Ethernet Verbindung”*.

<sup>1</sup><http://www.ti.com/tool/ccstudio/>, Zugriffsdatum:11.03.2018

<sup>2</sup><http://www.ti.com/tool/halcodegen/>, Zugriffsdatum:11.03.2018

<sup>3</sup><https://www.ixxat.com/products/products-industrial/pc-interfaces/pc-can-interfaces/details-pc-can-interfaces/usb-to-can-v2/>, Zugriffsdatum:11.03.2018

## 6. Praktische Arbeit

### 6.2.2. Softwareentwicklung

Die Software wurde in der CCS-Entwicklungsumgebung entwickelt und getestet. Als erstes wurden mit Hilfe von HALCoGen grundlegende und optionale Treiber (z.B. Ethernet-, CAN-, SCI-Treiber) für die Inbetriebnahme der Evaluierungsboards konfiguriert und erstellt. Diese generierten Treiber-Dateien wurden daraufhin in ein erstelltes CCS-Projekt importiert.

Um eine Ethernet-Funktion zu ermöglichen, wurde der *lwIP-TCP/IP-Stack* (Abschn. 6.2.2.1) in das Projekt eingebunden. Dieser enthält alle Funktionen, die für eine Ethernet-Kommunikation notwendig sind. Für den Austausch von Ethernet-Nachrichten wurden die UDP (Abschn. 2.1.1.1)- und TCP (Abschn. 2.1.1.2)-Protokolle verwendet. Die TCP und UDP Initialisierung, wie auch die Funktionen für Versenden und Empfangen, befinden sich in den `source\sys_udp.c` und `sys_tcp.c` Dateien.

Die CAN1-Module wurden für das Empfangen und Versenden von CAN-Nachrichten konfiguriert. Die Konvertierung von CAN- in Ethernet-Nachrichten wurde so gelöst, dass das Datensegment der CAN-Nachricht entpackt, ins Datensegment (Payload) der Ethernet-Nachricht verpackt und daraufhin verschickt wird. An der anderen Seite wird die Nachricht entpackt, das Datensegment anschließend in eine CAN-Nachricht verpackt und aus der CAN-Schnittstelle rausgeschickt.

Es wurden verschiedene Optionen erstellt, welche die Konfiguration des Demonstrators erleichtern und optionale Betriebs-Funktionen ermöglichen:

- **Master und Slave:** Diese Option verleiht jedem Evaluierungsboard eine eigene MAC- und IP-Adresse. Wenn beide Evaluierungsboards als *Master* oder *Slave* definiert werden, wird die Verbindung nicht hergestellt. Die Definition befindet sich in der `lwip-1.4.1\ports\hdk\include\netif\phy_dp83640.h` Datei (Abb. 6.4).
- **UDP oder TCP:** Die Daten können entweder, als eine UDP- oder TCP-Nachricht verschickt werden. Die Definition für beide Protokolle befindet sich in der `include\sys_udp.h` Datei. Bei Definition von TCP muss in der `include\sys_tcp.h` Datei jedes Evaluierungsboard entweder als *Client* oder *Server* definiert werden.
- **CAN- oder Terminal-Ausgang:** Mit dieser Option wird definiert, ob die empfangenen UDP- oder TCP-Nachrichten als eine CAN-Nachricht über die CAN1-Schnittstelle rausgeschickt oder direkt im Terminal-Fenster angezeigt werden. Die Definition befindet sich in der `include\sys_udp.h` Datei.
- **Test-Mode:** Mit der *Test-Mode*-Funktion können beliebig viele Ethernet-Pakete, die mit zufälligen Daten befüllt werden, generiert und gesendet werden. Diese Funktion ist bei einer *Bandwidth-Messung* hilfreich. Die Test-Mode Funktion `testMode()` befindet sich in `source\sys_main.c` und wird in der `include\sys_udp.h` Datei definiert.

## 6.2.2.1. Design und Initialisierung vom lwIP

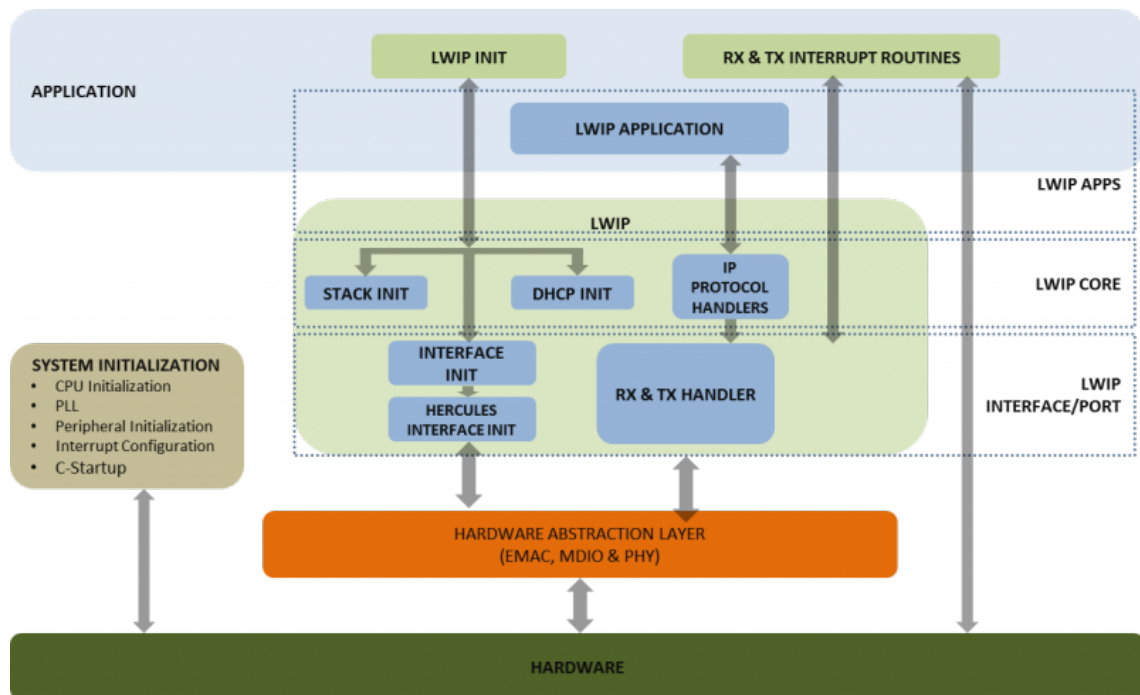


Abbildung 6.3.: Design der lwIP Implementation[[14]]

Der Implementierte lwIP-TCP/IP-Stack (Abb. 6.3) besteht aus 4 Schichten:

1. **Hardware-Abstraktionsschicht** (*Hardware Abstraction Layer*): Besteht aus dem *Ethernet Media Access Controller* (EMAC), *Management Data Input/Output* (MDIO) und dem PHY. EMAC und MDIO sind Teil des generierten HALCoGen-Codes und stellen eine voll funktionsfähige Ethernet-Schnittstelle, nach dem *IEEE 802.3*-Standard, bereit.

In der `source\emac.c` Datei, befinden sich die Funktionen des EMAC-Moduls und in der `source\mdio.c` die Funktionen der MDIO-Schnittstelle. Die `lwip-1.4.1\ports\hdk\include\netif\phy_dp83640.c` und `phy_bcm89810.c` Datei beinhalten benötigte Funktionen der Ethernet-PHYs DP83640 und BCM89810. PHY ist Teil des lwIP-Pakets[[14]].

2. **lwIP Netzwerk-Interface-Schicht:** Repräsentiert die Netzwerkgerät (NetIf)-Schnittstelle zum lwIP TCP/IP-Stack. Um mit dem Netzwerk kommunizieren zu können, muss die Hardware-Abstraktionsschicht mit einem TCP/IP-Stack (lwIP) zusammengeführt werden, welcher die Netzwerk-Pakete verwalten kann. Die Hardware-Abstraktionsschicht bindet sich mit der Interface-Schicht des lwIP und stellt das gerätespezifische Interface zum lwIP dar.

## 6. Praktische Arbeit

Das Netzwerkgerät (*network device*) ist von einem *struct netif* repräsentiert. Das *netif* enthält alle Informationen über das Interface z.B. IP/MAC Adressen, TCP/IP-Optionen, Link-Informationen und Protokoll-handler. Alle benötigten Funktionsaufrufe für die Initialisierung des lwIP-Stacks und Netzwerk-Interfaces befinden sich in `lwip-1.4.1\ports\hdk\lwiplib.c` [[14]].

3. **lwIP-Applikationsschicht:** Ist eine IP-Stack-Applikation, die auf LwIP basiert. Es werden Beispiele für HTTP-Server und UDP basierende Client- und Echo-Server zur Verfügung gestellt. Die Pakete starten und enden in dieser Schicht.
4. **System-Applikationsschicht:** Implementiert die System-Initialisierung (`lwip-1.4.1\ports\hdk\lwiplib.c`) (Abb. 6.4), welche die Oszillatoren (PLLs) und den Pin-mux konfiguriert und bietet Optionen für den lwIP-Stack (`include\lwipopts.h`).

Die Abb. 6.4 zeigt die Projektstruktur des Standard-Ethernet-Demonstrators.

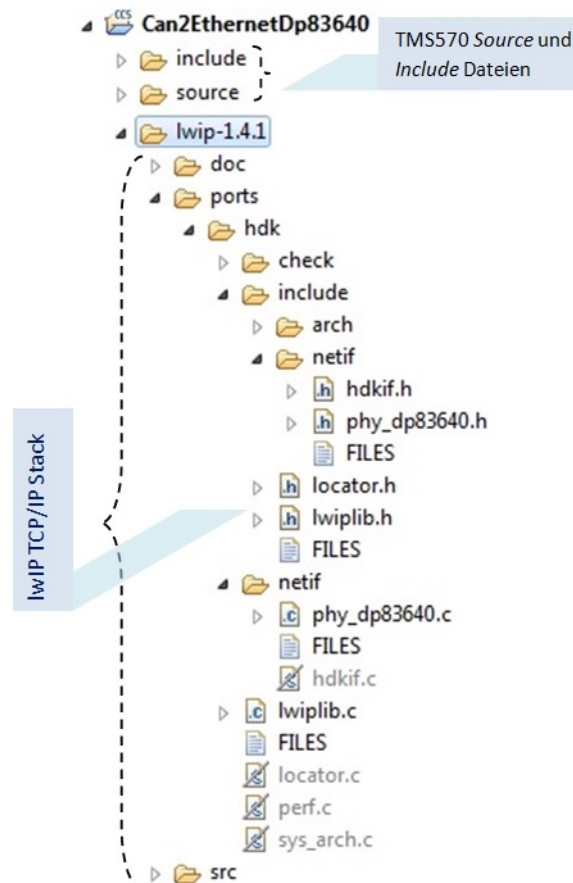


Abbildung 6.4.: Projektstruktur des Standard-Ethernet-Demonstrators

Die Netzwerkgerät-Initialisierung wird in der *lwIPInit()* Funktion aufgerufen (`lwip-1.4.1\ports\hdk\lwiplib.c`), nachdem das Netzwerkgerät im lwIP-Stack mittels *netif\_add()* registriert wurde. Die eigentliche Hardware-Initialisierung passiert in der *hdkif\_init()* Funktion, welche dann die *hdkif\_hw\_init()* Funktion für die *low-level*-Initialisierung aufruft. Die Funktionen für die Netzwerkgerät-Initialisierung befinden sich in der `lwip-1.4.1\ports\hdk\netif\hdkif.c` Datei[[14]].

Die lwIP-Konfigurationsdatei befindet sich in `include\lwipopts.h`. Weitere Optionen können per *copy/paste* aus der `lwip-1.4.1\src\include\lwip\opt.h` Datei importiert werden.

### 6.2.3. Aufbau und Inbetriebnahme

Die Ethernet-Schnittstellen der zwei Evaluierungsboards werden mit einem Standard-Ethernet-Kabel miteinander verbunden. An die CAN1-Schnittstelle wird ein CAN-Analyser, in unserem Fall der *USB-to-CAN-Compact* von IXXAT, angeschlossen.

Der *S2 DIP Switch* am Evaluierungsboard muss auf *ETHERNET ON* gesetzt werden. Diese Einstellung ermöglicht die Kommunikation des integrierten PHY (Dp83640) mit dem Mikrocontroller (MCU). Beim BroadR-Reach-Demonstrator muss dieser ausgeschaltet werden, wodurch der PHY (Dp83640) isoliert wird und das Multiplexen der MII-Signale auf die Steckleisten ermöglicht wird.

Die Evaluierungsboards werden über die *JTAG USB*-Schnittstelle an den Rechner angeschlossen. Der Demonstrator befindet sich in einem kompakten Gehäuse, wo alle Schnittstellen leicht zugänglich sind. Die Evaluierungsboards werden mit 9 V Batterien betrieben. Die Abb. 6.5 zeigt den Aufbau des Standard-Ethernet-Demonstrators.



## 6. Praktische Arbeit



Abbildung 6.5.: Aufbau des Standard-Ethernet-Demonstrators

Nach dem Aufbau des Demonstrators muss die entwickelte Software auf die Evaluierungsboards aufgespielt werden. Das Konfigurieren und Aufspielen der Software wird in CCS gemacht. Als erstes werden die verfügbaren Optionen (siehe Abschn. 6.2.2) nach Bedarf konfiguriert und anschließend auf die Evaluierungsboards aufgespielt.

Das Versenden und Empfangen der CAN-Nachrichten über die CAN-Analyser wird mit Hilfe des MiniMon-Programms durchgeführt. Diese Nachrichten werden in das CAN-Modul reingeschickt, daraufhin als eine UDP oder TCP Nachricht verpackt und aus der Ethernet-Schnittstelle zum anderen Board weiter geschickt. Dort wird die empfangene Nachricht entpackt und als eine CAN-Nachricht aus dem CAN-Modul wieder rausgeschickt. Diese Nachricht kann dann im MiniMon betrachtet werden. Dieser Datenfluss kann in beide Richtungen gleichzeitig ausgeführt werden, also Voll-Duplex.

Eine detaillierte Beschreibung der Inbetriebnahme und Verwendung der Programme, kann dem separaten Dokument *„Dokumentation - CAN to Ethernet Verbindung“* entnommen werden. In der Abb. 6.6 sieht man das Terminal-Fenster des CCS, welches den 100 Mbit/s Verbindungs-Aufbau des Standard-Ethernet-Demonstrators und den Datenaustausch zwischen den Evaluierungsboards anzeigt.

## 6.2. Demonstrator - Standard Ethernet

The screenshot displays the MiniMon V3 by IXXAT interface, which is used for monitoring CAN and Ethernet communication. The interface is divided into several sections:

- IXXAT Interfaces:** Shows the status of various interfaces:
  - USB-to-CAN compact: 00:24:44:2421
  - USB-to-CAN compact: 00:24:47:640
  - CAN1: SJA1000
  - USB-to-CAN compact: 00:30:49:405
- Controller Status:**
  - Controller initialized
  - Low speed transceiver
  - Transmit pending
  - Data overrun
  - Error warning level
  - Bus off
  - Baudrate: 500 kbit/s
  - Busload %
- Message Log (Tx):**

Identifier	Ext.	Rtr	Data	Cycle Count	Cycle Time (ms)	Cycle Mode
40 41 52 48 4D	<input type="checkbox"/>	<input type="checkbox"/>	40 41 52 48 4D	1000	0	None
40 41 52 48 4D	<input type="checkbox"/>	<input type="checkbox"/>	40 41 52 48 4D	0	0	None
40 41 52 48 4D	<input type="checkbox"/>	<input type="checkbox"/>	40 41 52 48 4D	0	0	None
40 41 52 48 4D	<input type="checkbox"/>	<input type="checkbox"/>	40 41 52 48 4D	0	0	None
- Terminal/Console:**

```

Result of transmission: The operation completed successfully. (0x0)

CAN TO ETHERNET COMMUNICATION with 0983646

Initializing Ethernet (Static IP)
DEBUG - Getting PHY ID...SUCCESS
DEBUG - Getting PHY Alive Status...SUCCESS
DEBUG - Setting up Link...SUCCESS
DEBUG - Setting up Link...SUCCESS
..DONE
Device IP Address: 192.168.0.2
<<Receive data on Ethernet via UDP: 192.168.0.2>>
<<Send data out via UDP: 192.168.0.2>>
<<Send data over Ethernet via UDP>>

```

Abbildung 6.6.: Verbindungs-Aufbau und Daten-Übertragung: Nachrichten werden in das CAN-Modul reingeschickt, daraufhin als eine UDP oder TCP Nachricht verpackt und aus der Ethernet-Schnittstelle zum anderen Board weiter geschickt. Da wird die empfangene Nachricht entpackt und als eine CAN-Nachricht aus dem CAN-Modul wieder raus geschickt.

## 6. Praktische Arbeit

### 6.3. Demonstrator - Automotive Ethernet

Beim Automotive-Ethernet-Demonstrator sollte die Standard-Ethernet-Implementierung (siehe Abschn. 6.2) durch die automobiltaugliche BroadR-Reach®-Technologie (siehe Abschn. 5.4.1) erweitert werden.

Um die Standard-Ethernet-Technologie auf die Automotive-BroadR-Reach-Technologie umzustellen, wurde die komplette physikalische Schicht des Standard-Ethernet neu entworfen. Dafür wurden zwei auf dem BroadR-Reach-Physical-Layer basierten Platinen entworfen.

Bei dem verwendeten Transceiver handelt es sich um den *100 Mbit/s Single-Port BCM89810<sup>4</sup>-Transceiver* des Unternehmen Broadcom, welcher auf der BroadR-Reach-Technologie basiert. Die im Standard-Ethernet-Demonstrator (Abschn. 6.2) verwendeten Transceiver (Dp83640), wurden von der Kommunikation isoliert.

Die Abb. 6.7 zeigt den Signalfluss des BroadR-Reach-Demonstrators. Dieser ist ähnlich wie beim ersten Demonstrator (siehe Abb. 6.1).

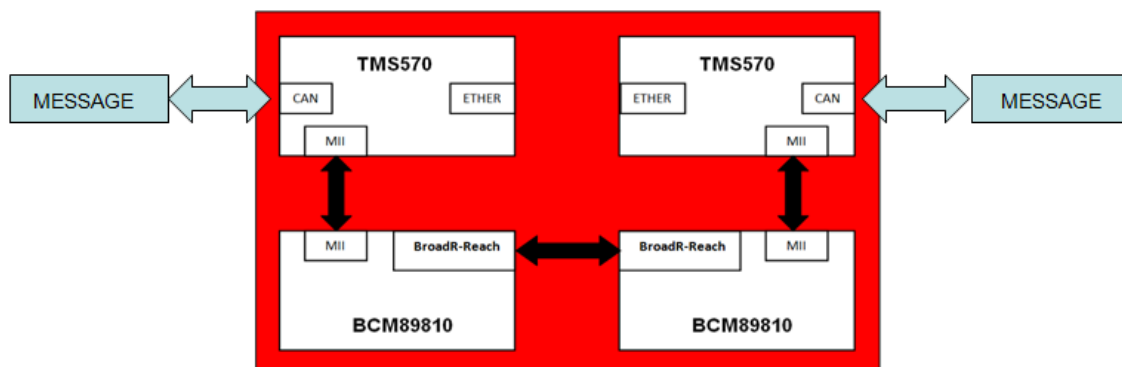


Abbildung 6.7.: Datenfluss des BroadR-Reach-Ethernet Demonstrators

Der Unterschied ist, dass die Daten nicht zwischen den integrierten Ethernet-Schnittstellen auf den Evaluierungsboards (Abb. 6.2) ausgetauscht werden, sondern zwischen den mit der MII-Schnittstelle verbundenen BroadR-Reach-Platinen.

Anschließend wurden am aufgebauten Demonstrator standardmäßige Automotive-EMV-Messungen vorgenommen und es wurde die Signalqualität bei verschiedenen Kabellängen gemessen.

<sup>4</sup><https://www.broadcom.com/products/ethernet-connectivity/copper-phy/automotive/bcm89810/>, Zugriffsdatum: 11.03.2018



## 6.3.1. Entwurf und Erstellung der Platine

Die Platine wurde mit der *EAGLE PCB Design Software*<sup>5</sup> entworfen.

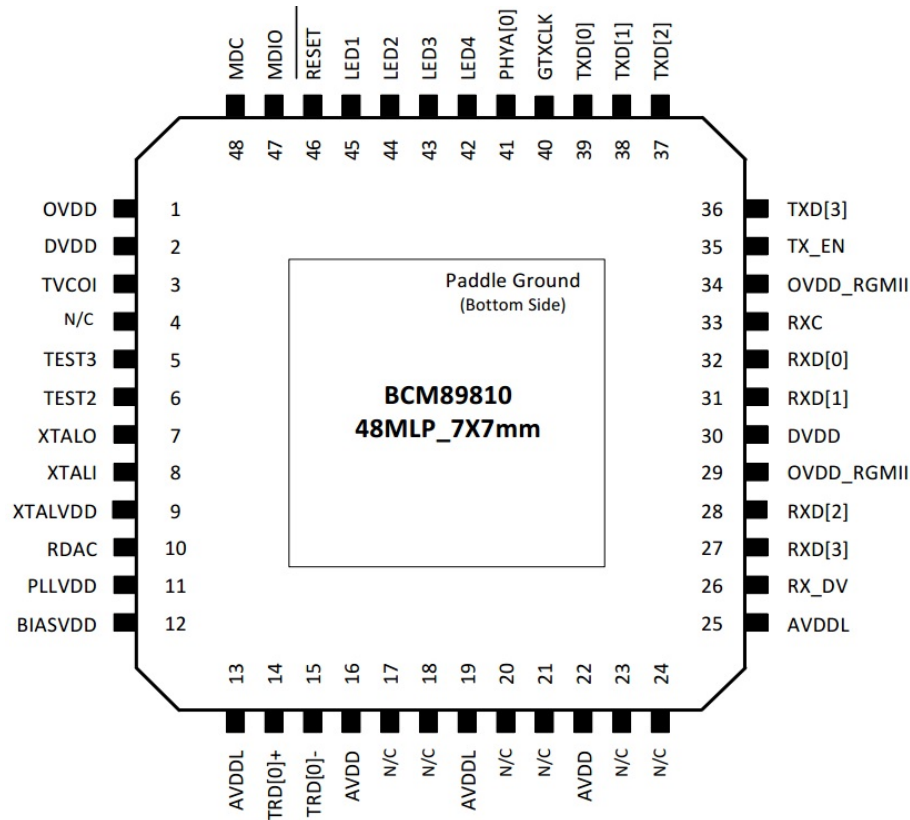


Abbildung 6.8.: Pinout Diagramm des BCM89810-Transceiver

Beim Entwurf des Layouts wurde darauf geachtet, dass man die EMV-Richtlinien des Herstellers einhält. Die verwendeten Bauelemente wurden dem Datenblatt und einem Prototyp-Schaltplan entnommen. Eine Liste der verwendeten Bauelemente befindet sich im Anhang (Abb. A.1). Die Platine wurde 4-lagig entworfen, bei welcher die Signalleitungen und die Versorgungsleitungen voneinander möglichst weit getrennt wurden.

Das Layout ist an die Steckleisten (*Multiplexed Pins*) (Abb. 6.2) des Evaluierungsboards angepasst, wo sich alle Multiplex-Signale der MII-Schnittstelle befinden. Da sich diese Signale unterschiedlich entfernt auf den Steckleisten befinden, war es nicht bei allen möglich den optimalen Leitungsweg zu wählen.

<sup>5</sup><http://www.cadsoft.de/eagle-pcb-design-software/?language=de/>, Zugriffsdatum: 11.03.2018

## 6. Praktische Arbeit

Die Platinen wurden am Institut für Elektronik<sup>6</sup> der Technischen Universität Graz bestückt und im Reflow-Lötofen gelötet. Die Unterseite der Platine, Konnektor und die Stiftleisten wurden händisch gelötet. Die Abb. 6.9 zeigt Ober- und Unterseite der fertigen Platine und die Beschreibung der einzelnen Bauelemente.

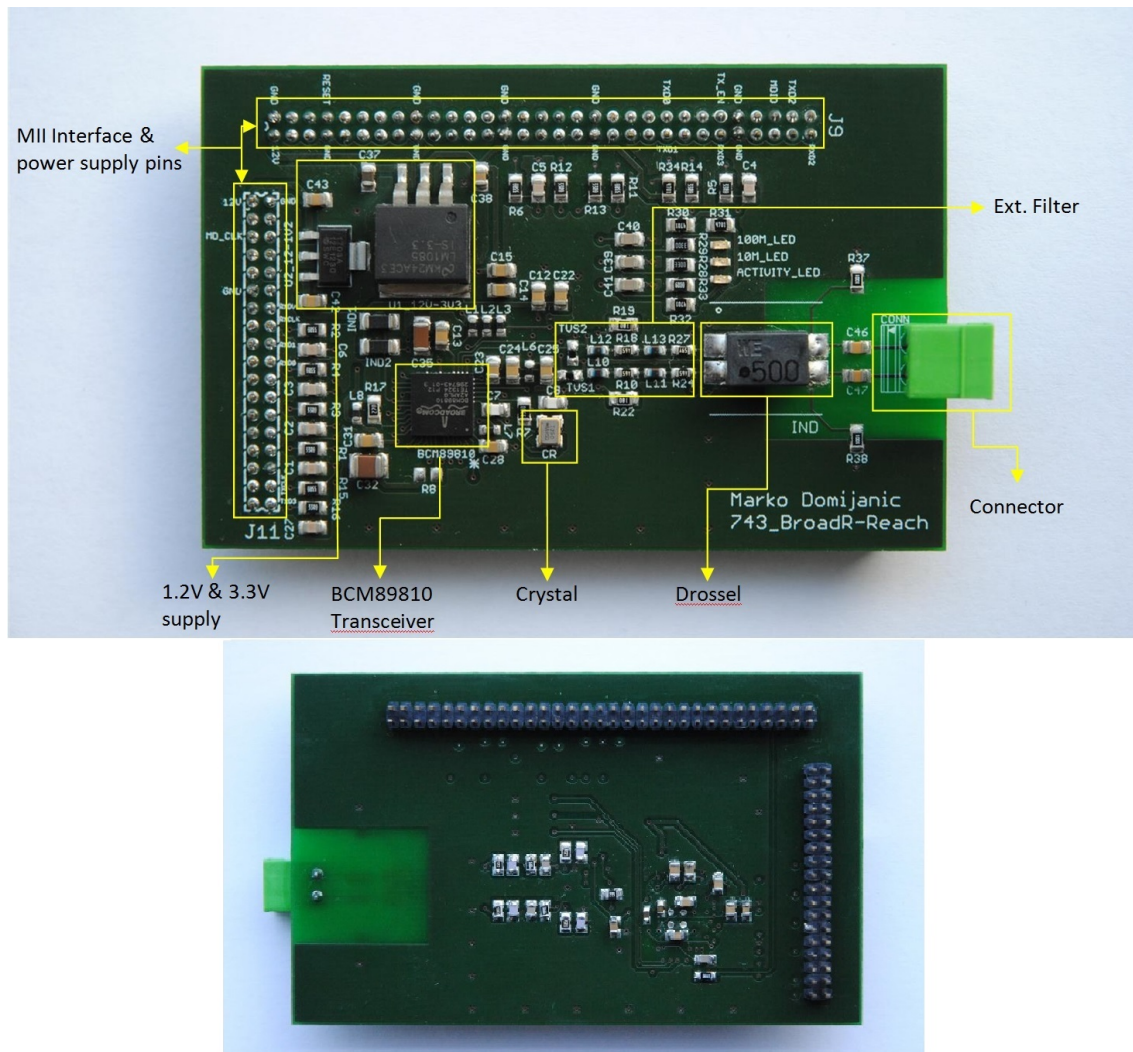


Abbildung 6.9.: Ober- und Unterseite der fertigen Platine

Die Abb. 6.10 zeigt den fertigen Schaltplan und die Abb. 6.11 das Layout der Platine.

<sup>6</sup><http://www.ife.tugraz.at/>, Zugriffsdatum: 11.03.2018

### 6.3. Demonstrator - Automotive Ethernet

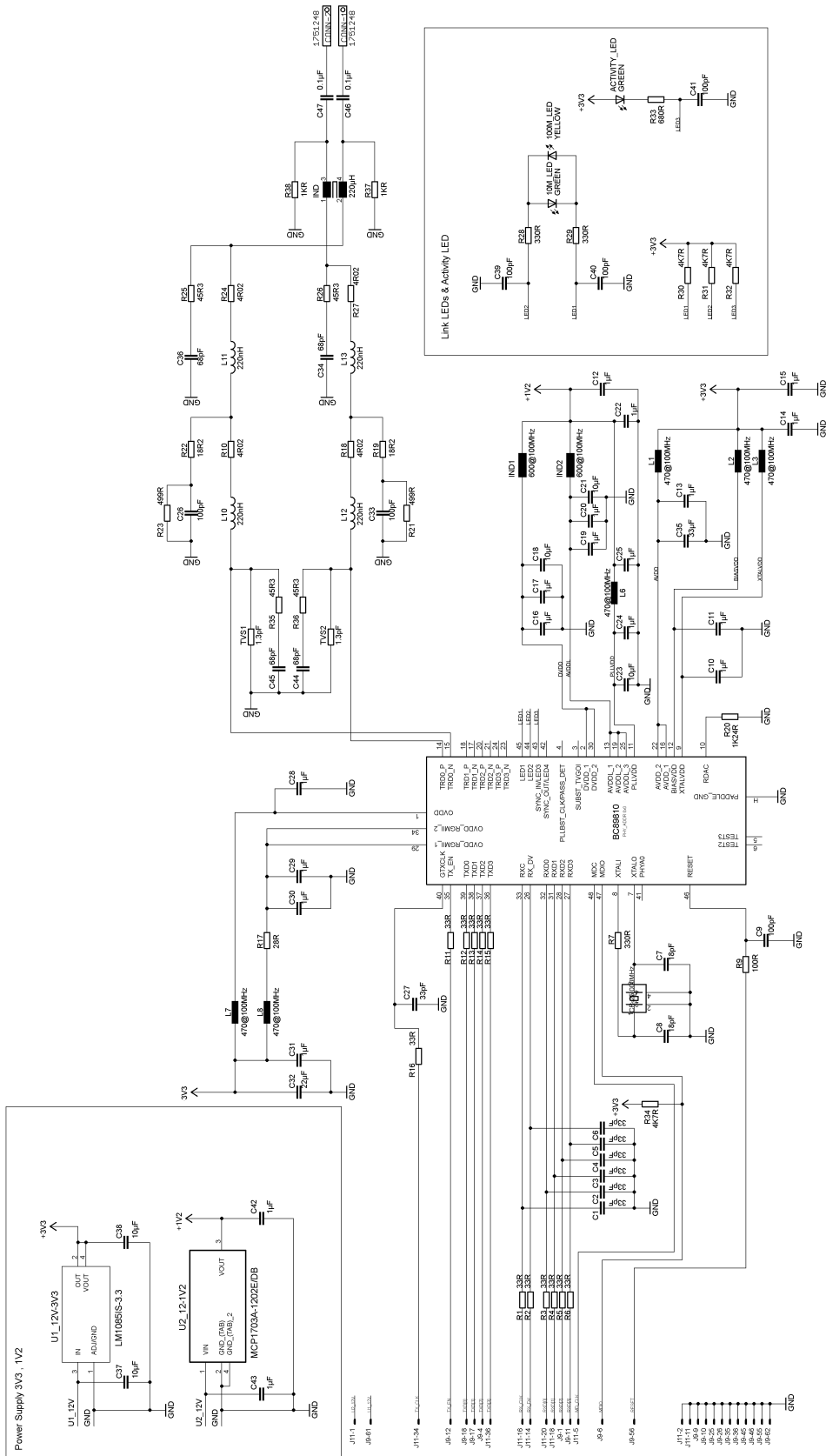


Abbildung 6.10.: Schaltplan der entworfenen Platine

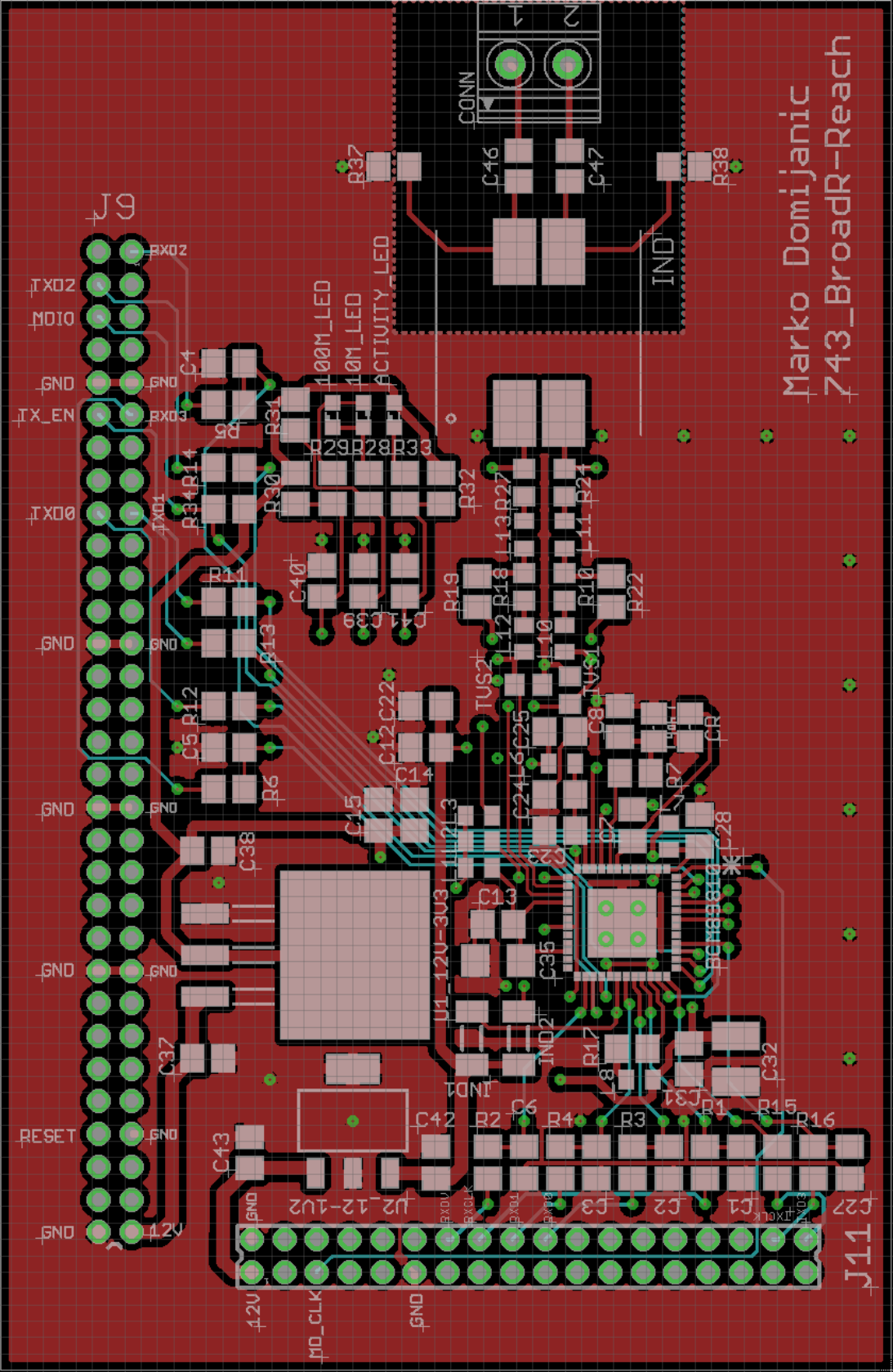


Abbildung 6.1.1.: Layout der entworfenen Platine

## 6.3.2. Software-Erweiterung

Um die entworfenen BroadR-Reach-Platinen in Betrieb zu nehmen, wurde die vorhandene Software aus dem Standard-Ethernet-Demonstrator erweitert. Als erstes ist der lwIP-Stack um die BCM89810-Register-Adressen und die nötigen Funktionen für den Zugriff auf jene Register erweitert worden.

Um die MII-Schnittstelle verfügbar zu machen, wurde der *S2-DIP-Switch* am Evaluierungsboard auf *ETHERNET OFF* gesetzt. Dieser isoliert den Integrierten PHY (Dp83640) vom MII-Interface des MCU und ermöglicht damit das Multiplexen der MII-Signale auf die Steckleisten. Danach können die Register des BCM89810-Transceiver über die MDIO-Schnittstelle geschrieben und gelesen werden.

Das komplette MII-Signal-Set besteht aus 16 Signalen. Der BCM89810-Transceiver bietet einen *MII-Lite- Modus*, in welchem einige Signale ausgesetzt werden können. Dieser Modus verzichtet auf das TX\_ER, RX\_ER, COL und CRS Signal und ist hilfreich im 100 Mbit/s BroadR-Reach Betrieb[[15]]. Der Demonstrator arbeitet im MII-Lite Modus.

Die Abb. 6.12 zeigt die einzelnen MII-Signale zwischen der Evaluierungsboard- und PHY-MII-Schnittstelle im MII-Lite Modus. Die nicht verwendeten Signale im MII-Lite wurden Rot durchgestrichen.

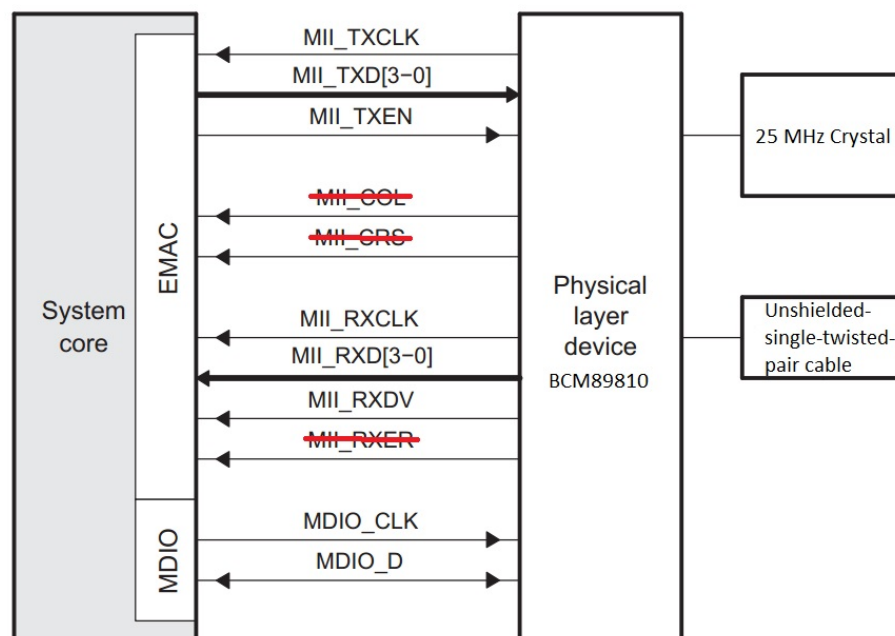


Abbildung 6.12.: Die verwendeten MII-Signale im MII-Lite-Modus zwischen der Evaluierungsboard- und PHY-MII-Schnittstelle[[15]]

## 6. Praktische Arbeit

BroadR-Reach verwendet ein neues *Autonegotiation-Verfahren* namens *Long Distance Signaling (LDS)*, um die Verbindungsparameter festzusetzen. Die Link-Partner entscheiden untereinander welcher der *Master* und welcher der *Slave* ist und wählen die höchst mögliche Übertragungsrate. LDS ist standardmäßig im BCM89810-Register gesetzt.

Der restliche *Software-Code* für die komplette Hardware-Initialisierung und die Übertragungsfunktionen wurden vom Standard-Ethernet-Demonstrator übernommen (siehe Abschn. 6.2.2).

### 6.3.3. Aufbau und Inbetriebnahme

Die entworfenen Platinen (Abb. 6.9) werden jeweils mit den Steckleisten (*Multiplexed Pins*) der Evaluierungsboards verbunden, wo alle Signale der MII-Schnittstelle sowie die Spannungsversorgungs-Pins vorhanden sind.

Mit einem ungeschirmten, verdrehten Zweidraht-Kabel (Abb. 6.13) werden die Platinen miteinander verbunden.



Abbildung 6.13.: Ungeschirmtes-verdrilltes-Zweidraht-Kabel

Über den *S2-DIP-Switch* am Evaluierungsboard wird der Integrierte PHY (Dp83640) vom MII-Interface des MCU Isoliert und ermöglicht so das Multiplexen der MII-Signale auf die Steckleisten.



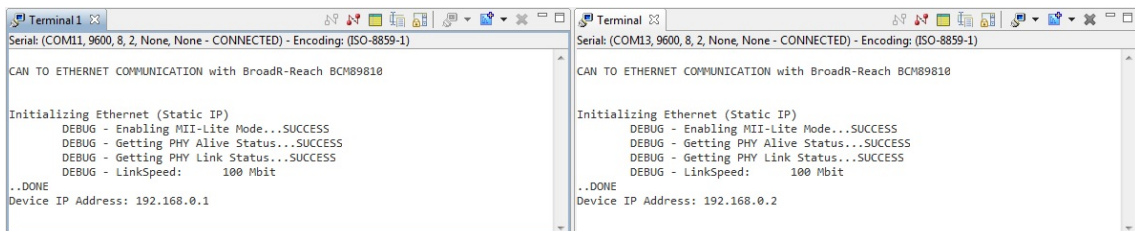
Der Demonstrator befindet sich in einem kompakten Gehäuse, wo alle Schnittstellen leicht zugänglich sind. Die Evaluierungsboards werden mit 9 V Batterien betrieben, daher kann eine einfache Funktions-Demonstration z.B. im Demo-Modus ohne zusätzliche Kabel oder einen Rechner durchgeführt werden.

Alle anderen Schritte bezüglich Aufbau bleiben gleich wie beim Standard-Ethernet-Demonstrator (siehe Abschn. 6.2.3).

Der einzige Unterschied in der Software-Inbetriebnahme, im Gegensatz zum Standard-Ethernet-Demonstrator, ist die Definition für das *Master-* und *Slave-Board*. Diese befindet sich in der `lwip-1.4.1\ports\hdk\include\netif\phy_bcm89810.h` Datei (Abb. 6.4). Die restlichen Schritte sind gleich wie beim Standard-Ethernet-Demonstrator (siehe Abschn. 6.2.3).

Nach dem Verbinden der BroadR-Reach-Platinen werden die Verbindungsparameter über das LDS-Verfahren untereinander ausgemacht. Nach dem erfolgreichen 100 Mbit/s Verbindungsaufbau, der in der Regel 200 ms dauert, startet je nach ausgewähltem Modus (siehe Abschn. 6.2.2) der Betrieb.

In der Abb. 6.14 sieht man den Aufbau der 100 Mbit/s BroadR-Reach-Verbindung im Terminal-Fenster.



```
Terminal1
Serial: (COM11, 9600, 8, 2, None, None - CONNECTED) - Encoding: (ISO-8859-1)
CAN TO ETHERNET COMMUNICATION with BroadR-Reach BCM89810
Initializing Ethernet (Static IP)
DEBUG - Enabling MII-Lite Mode...SUCCESS
DEBUG - Getting PHY Alive Status...SUCCESS
DEBUG - Getting PHY Link Status...SUCCESS
DEBUG - LinkSpeed: 100 Mbit
..DONE
Device IP Address: 192.168.0.1

Terminal
Serial: (COM13, 9600, 8, 2, None, None - CONNECTED) - Encoding: (ISO-8859-1)
CAN TO ETHERNET COMMUNICATION with BroadR-Reach BCM89810
Initializing Ethernet (Static IP)
DEBUG - Enabling MII-Lite Mode...SUCCESS
DEBUG - Getting PHY Alive Status...SUCCESS
DEBUG - Getting PHY Link Status...SUCCESS
DEBUG - LinkSpeed: 100 Mbit
..DONE
Device IP Address: 192.168.0.2
```

Abbildung 6.14.: Aufbau der 100 Mbit/s BroadR-Reach-Verbindung im Terminal-Fenster

## 6. Praktische Arbeit

Die Abb. 6.15 zeigt den fertigen Aufbau und die Inbetriebnahme des BroadR-Reach-Demonstrators im 100 Mbit/s Betrieb. In einem kurzen Funktions-Video<sup>7</sup> kann man den Aufbau der Verbindung und eine kurze Übertragung betrachten.



Abbildung 6.15.: Aufbau und Inbetriebnahme des BroadR-Reach-Demonstrators im 100 Mbit/s Betrieb.

<sup>7</sup><https://vimeo.com/114179302/>, Zugriffsdatum: 11.03.2018 Passwort: ethernet



### 6.3.4. Signalqualität Messungen

In dieser Untersuchung wurde die Signalqualität einer 100 Mbit/s Verbindung, bei verschiedenen Kabellängen zwischen den beiden BroadR-Reach-Platinen, gemessen. Dafür wurden Kabellängen von 1, 3, 5, 10, 15 und 20 Meter verwendet.

Der BCM89810-Transceiver bietet einen *Signal Quality Indicator* (SQI), mit welchem man über einen Algorithmus die Signalstärke ausrechnen kann. Das Ergebnis der Berechnung ist ein Wert von 0 bis 10. Tabelle 6.2 zeigt die Beschreibung der einzelnen SQI-Werte.

Tabelle 6.2.: Beschreibung der einzelnen SQI-Werte

<i>SQI-Wert</i>	<i>Beschreibung</i>
0-2	Fehler treten auf
2-4	Schlecht
4-6	Genügend
6-8	Gut
8-10	Ausgezeichnet

Die Differenz zwischen den einzelnen Werten beträgt ungefähr 1 dB. Ein Wert unter 2 bedeutet, dass keine funktionierende BroadR-Reach-Verbindung aufgebaut werden kann. Die Abb. 6.16 zeigt ein Diagramm der gemessenen SQI-Werte bei verschiedenen Kabellängen und die Tabelle 6.3 die genauen gemessenen Werte.

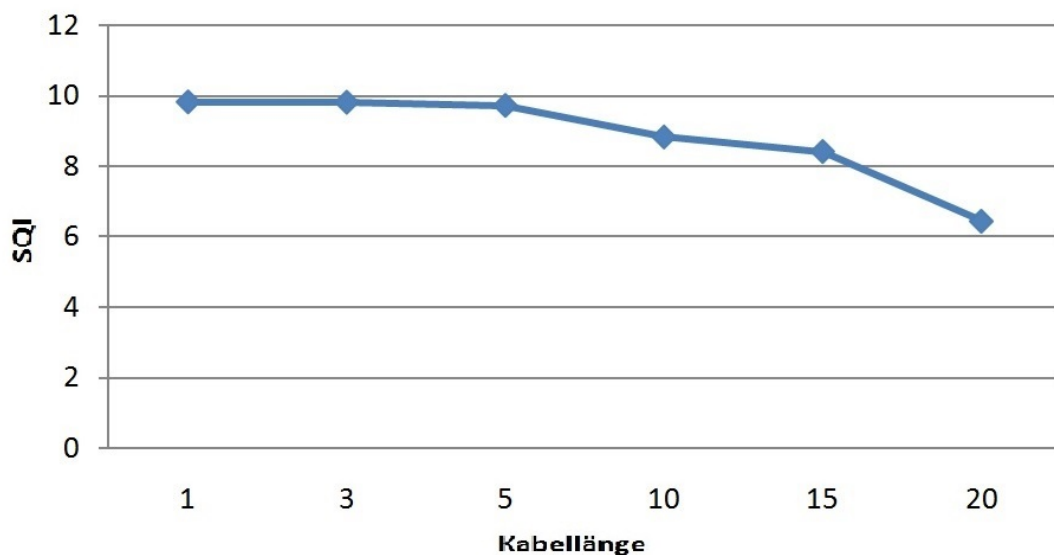


Abbildung 6.16.: Signalqualität bei verschiedenen Kabellängen

## 6. Praktische Arbeit

Tabelle 6.3.: SQI Werte bei verschiedenen Kabellängen

Kabellänge	1 m	3 m	5 m	10 m	15 m	20 m
SQI	9,81	9,8	9,71	8,82	8,41	6,44

Wie man aus den Ergebnissen der Messung sehen kann, verschlechtert sich die Signalqualität mit größerer Kabellänge. Die beste Signalqualität ist bis ungefähr 15 Meter, was für den Einsatz im Kfz auch ausreichend ist. Danach nimmt sie in größeren Schritten ab.

### 6.3.5. EMV Untersuchung

Im Vergleich zum Consumer-Bereich herrschen in der Automobil Umgebung höhere Anforderungen an elektronische Systeme und ihre Komponenten.

Aus Kosten- und Gewichtsgründen muss bei Ethernet auf eine Schirmung der Leitungen und der Steckverbinder verzichtet werden. Dies und die hohe Übertragungsrate von 100 Mbit/s stellen eine große Herausforderung an die elektromagnetische Verträglichkeit (EMV) dar. Bei einer ungünstigen Auslegung des Systems kann es zum Fehlverhalten oder gar einem Ausfall kommen.

Systeme mit einen BroadR-Reach® Transceiver müssen in der Lage sein die EMV Anforderungen für den Automobilbereich einzuhalten. Grenzwerte und Prüfverfahren sind in den Normen CISPR 25<sup>8</sup> und ISO 11452-2&4<sup>9</sup> definiert.

Das Ziel dieser Untersuchung war die Prüfung der Störfestigkeit (SF) einer BroadR-Reach®-Verbindung. Als Störfestigkeit bezeichnet man die Fähigkeit eines elektronischen Systems, Störgrößen bestimmter Höhe ohne Fehlfunktion standzuhalten.

<sup>8</sup><https://webstore.iec.ch/publication/78/>, Zugriffsdatum: 11.03.2018

<sup>9</sup>[http://www.iso.org/iso/catalogue\\_detail.htm?csnumber=36570/](http://www.iso.org/iso/catalogue_detail.htm?csnumber=36570/), Zugriffsdatum: 11.03.2018

Die Abb. 6.17 zeigt den Aufbau einer leitungsgeführten HF-Störfestigkeitsprüfung. Der HF-Generator (Hochfrequenz-Generator) wirkt hierbei als Erzeuger eines elektromagnetischen Störfeldes, das wiederum auf die Signalleitung der BroadR-Reach®-Verbindung einwirkt und dort leitungsgeführte Störungen hervorruft.

Untersucht wird, wie sich dieses elektromagnetische Störfeld auf die Signalqualität (SQI) (Abschn. 6.3.4) der BroadR-Reach®-Verbindung auswirkt. Bei einem SQI Wert unter zwei kann keine funktionierende BroadR-Reach®-Verbindung mehr aufgebaut werden.

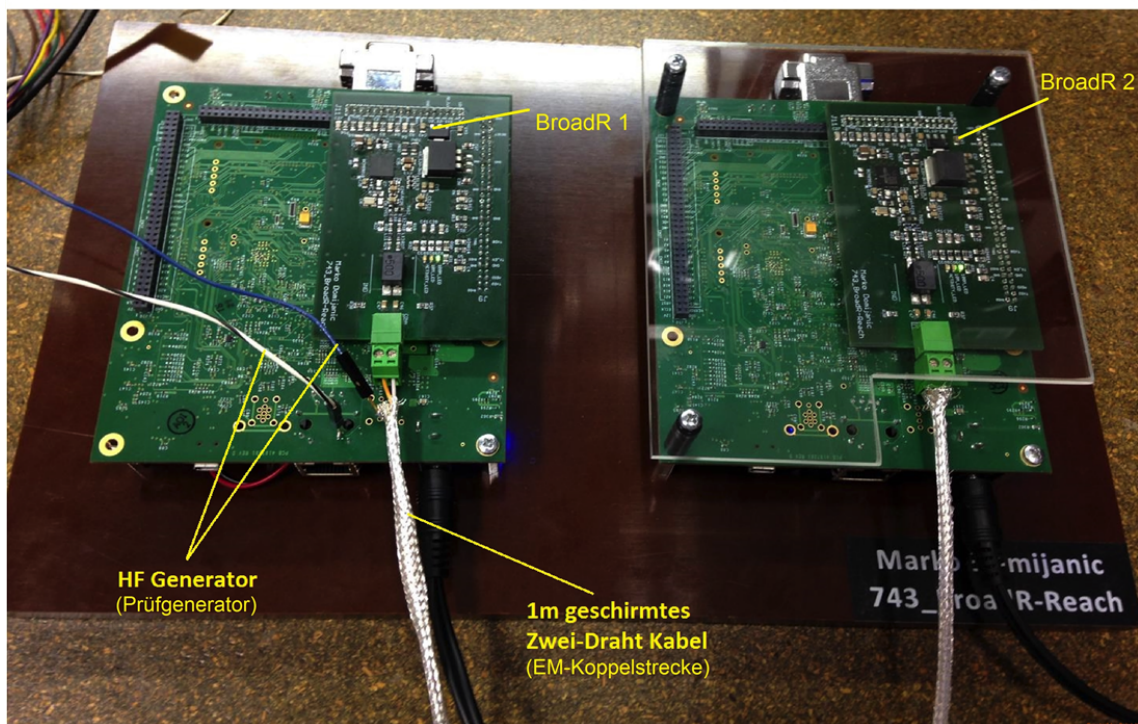
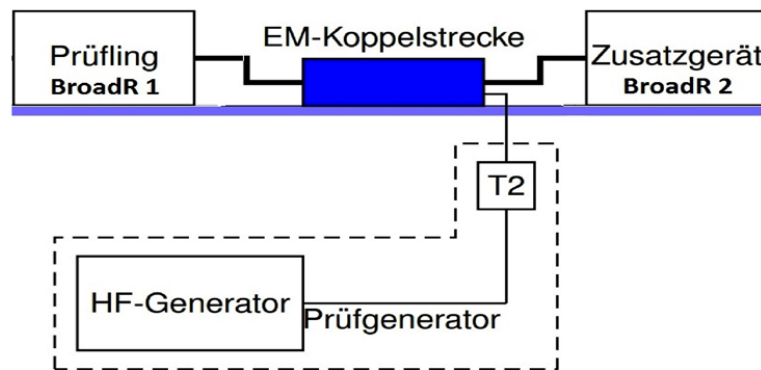


Abbildung 6.17.: Aufbau einer leitungsgeführten HF-Störfestigkeitsprüfung

## 6. Praktische Arbeit

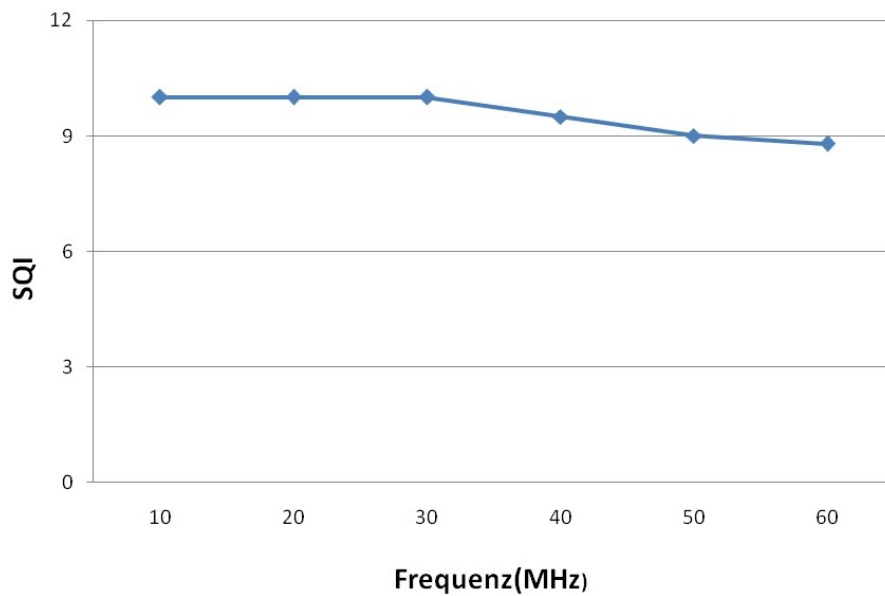


Abbildung 6.18.: Veränderung der Signalqualität (SQI) durch Erhöhung der Störgröße (MHz)

Wie man aus dem Diagramm (Abb. 6.18) entnehmen kann, verringert sich die Signalqualität (SQI) mit der Erhöhung der Störgröße (bis 50 MHz) minimal und führt zu keinem Ausfall oder Beeinträchtigung der Kommunikation.

Daher lassen die Untersuchungen darauf schließen, dass die BroadR-Reach® Technologie die Anforderungen in Bezug auf elektromagnetische Störfestigkeit in der Automobil-Umgebung erfüllt.

Ein kurzes Funktions-Video<sup>10</sup> zeigt den Aufbau und die Durchführung der Messung.

<sup>10</sup><https://vimeo.com/133409109/>, Zugriffsdatum: 11.03.2018 Passwort: ethernet

## 7. Zusammenfassung

Die immer wachsende Anzahl von Steuergeräten, kamerabasierte Fahrerassistenz-, Infotainment- und Multimedia-Systeme, erhöhen die Anforderungen an Vernetzungstechnologien im Fahrzeug. Vorhandene Bussysteme sind diesen Anforderungen in Bezug auf Bandbreite, Wirtschaftlichkeit und Komplexität des Systems nicht mehr gewachsen. Ethernet erfüllt diese Anforderungen und bietet sich jetzt als kostengünstige und verbreitete Lösung für den Einsatz im Kraftfahrzeug an.

Ziel dieser Masterarbeit war es eine umfassende Untersuchung des Themas „automotive Ethernet“ durchzuführen. Dabei sollte die Motivation hinter der Einführung ins Fahrzeug beschrieben, und eine Faktensammlung zum Stand der Technik von „automotive Ethernet“ durchgeführt werden.

Im praktischen Teil war ein Aufbau mit seriennahen automobiltauglichen Ethernet-Transceivern zu realisieren. Anschließend sollte die Einsatztauglichkeit von Ethernet im Kraftfahrzeug durch standardmäßige Untersuchungen (EMV, Signalqualität) bewertet werden.

Ethernet erleichtert den Einzug der Consumer Electronics (einfache Plug & Play Anbindung von Endgeräten und Applikationen) ins Fahrzeug. Damit wird sich das Fahrzeug in ein sogenanntes „Digital Home“ umwandeln, wo der Consumer die gleichen Internet- und Multimedia-Dienste nutzen kann wie in der Wohnung.

Die Haupttreiber von Ethernet sind zurzeit noch Multimedia Anwendungen und fortschrittliche kamerabasierte Fahrerassistenzsysteme (ADAS). Zu denen gehören Rückfahrkameras, Warnsysteme, Kollisionserkennung, Verkehrszeichenerkennung usw.

Da Ethernet eine bewährte und weit verbreitete Technologie ist, steht eine große Anzahl an verwendbaren Protokollen und Applikationen aus der PC- und Industrie-Ethernet-Welt zur Verfügung. Weiters sind viele Entwicklungs- und Diagnosewerkzeuge bereits vorhanden.

Aktuell ist die 100 Mbit/s BroadR-Reach® Technologie (Abschn. 5.4.1) de-facto Standard für „automotive Ethernet“. Diese ermöglicht eine 100 Mbit/s Übertragung über ungeschirmte *Single Twisted Pair Kabel* und bietet dadurch einen deutlichen Kosten- und Gewichtsvorteil. Der Interessenverband OPEN Alliance SIG, welchem sowohl Unternehmen aus der Automobilindustrie, wie auch aus der Unterhaltungselektronik- und IT-Branche angehören, setzt auf die BroadR-Reach® Technologie und arbeitet daran diese als offenen Standard für „automotive Ethernet“ zu etablieren.

Mit Hilfe von Ethernet wird sich die Netzwerkarchitektur im Fahrzeug aus heterogenen Netzwerken, bestehend aus proprietären Protokollen wie MOST und CAN, in hierarchische homogene Ethernet-Netzwerke umwandeln (Abb. 5.3). Diese neue Fahrzeug-Netzwerkarchitektur hilft nicht nur die Kosten und das Gewicht zu verringern, sondern

## 7. Zusammenfassung

vereinfacht auch die Kommunikation zwischen den verschiedenen Systemen in- und außerhalb des Fahrzeugs. Um diesen Wandel von heterogenen- in homogene-Netzwerke zu schaffen, muss Ethernet noch einige Hindernisse überwinden.

Für Videoübertragung sind 100 Mbit/s ausreichend, aber zu wenig um als Backbone im Fahrzeug zu agieren. Deshalb wurde die Entwicklung des Nachfolgers von *100 Mbit/s Twisted Pair Standards* für Übertragungsraten bis 1 Gbit/s über eine verdrehte Zwei-Draht-Leitung in die Wege geleitet. Prototypen sollen noch heuer oder spätestens Ende 2016 erhältlich sein.

Neben der immer steigenden Bandbreiten-Anforderungen, müssen auch die strengeren EMV-Vorgaben eingehalten, sowie die Kosten und das Gewicht (*single twisted pair only*) reduziert werden. Durch Power over Ethernet (PoE) können Geräte wie z.B. Kameras mit Strom versorgt, und so die Verkabelungskosten und das Gewicht im Fahrzeug weiter reduziert werden.

Eine weitere Herausforderung ist die vollständige Integration von Ethernet und der TCP/IP-Protokollsuite in die AUTOSAR Architektur.

Um Ethernet für sicherheitskritische Anwendungen fit zu machen, müssen geeignete Echtzeitprotokolle standardisiert werden. Ethernet wurde bereits durch den AVB-Standard (Abschn. 5.5.1) um Echtzeitprotokolle für die Übertragung von zeitkritischen Audio- und Videodaten erweitert. Durch die *Time-Sensitive Networking* (TSN) Protokolle (Abschn. 5.5.2), welche eine beträchtliche Reduktion der Übertragungsdauer ermöglichen, wird Ethernet auch für Fahrerassistenz-Systeme und sicherheitskritische Anwendungen eingesetzt werden können. Somit wird Ethernet in der Zukunft CAN und Flexray bei solchen Anwendungen ersetzen.

Erste Serienfahrzeuge, welche die BroadR-Reach® Ethernet-Technologie verwenden, fahren schon heute auf den Straßen, darunter der neue BMW X5, Jaguar XJ und Volkswagen Passat.

Bis zum Jahr 2017 werden nach Schätzungen alle Fahrerassistenzsysteme (ADAS) und der Infotainment-Bereich Ethernet verwenden. Das heißt dass Ethernet voraussichtlich MOST und LVDS aus den meisten Fahrzeugen verdrängen wird.

Laut vorhersagen von *Strategy Analytics*<sup>1</sup> wird der „automotive Ethernet“ Markt in den folgenden Jahren exponentiell wachsen, und im Jahr 2020 die Anzahl der Ethernet Ports per Fahrzeug, 15 bis 20 betragen. Für weitere Informationen betreffend des Marktpotenziales von Ethernet, siehe Abschn. 5.8.2. Ethernet wird sich in Zukunft als primäres Bordnetz im Fahrzeug durchsetzen und alle Domänen miteinander verbinden.

---

<sup>1</sup><https://www.strategyanalytics.com/>, Zugriffsdatum: 11.03.2018

In den kommenden Jahren werden die Begriffe Fahrzeug und Mobilität neu definiert. Das Fahrzeug selber nimmt, mit Hilfe von fortschrittlichen Fahrerassistenzsystemen, immer mehr die Kontrolle über die Fahrentscheidungen und versetzt den Fahrer in eine passive Rolle.

Solche autonomen und semi-autonomen Fahrzeuge haben eine Vielzahl von Sensoren, die eine große Menge von Straßen- und Fahrzeugdaten sammeln. Diese Daten müssen gleichzeitig transportiert und so schnell wie möglich verarbeitet werden, so dass die Systeme im Fahrzeug die notwendigen Fahrentscheidungen treffen können. Ethernet bietet die Bandbreite und Effizienz um diese Vision zu ermöglichen.

Im praktischen Teil sollten auf Basis zweier Evaluierungsplattformen, *CAN to Ethernet* Demonstratoren mit jeweils Standard-Ethernet- und automotive-Ethernet-Transceivern aufgebaut werden.

Beim ersten Standard-Ethernet-Demonstrator (Abschn. 6.2) wurde eine Schnittstelle zwischen CAN und Ethernet, hard- und softwaremäßig in Betrieb genommen. Diese Schnittstelle empfängt Nachrichten vom CAN Modul, setzt sie auf ein Ethernet taugliches Protokoll (UDP/TCP) um und sendet diese über den Ethernet-Transceiver zur Empfängerplatine. Die Empfängerplatine arbeitet den Signalfluss in der umgekehrten Richtung nochmals ab. Dieser Datenfluss findet in beide Richtungen gleichzeitig statt (Vollduplex).

Beim zweiten Demonstrator wurden die Standard-Ethernet-Transceiver durch seriennahe automotive-Ethernet-Transceiver (BroadR-Reach®) ersetzt und auf die vorhandene Hardware- und Softwarestruktur adaptiert. Um die Standard-Ethernet- auf die automotive-BroadR-Reach®-Technologie umzustellen, wurde die komplette physikalische Ethernet Schicht neu entworfen. Dafür wurden zwei auf dem BroadR-Reach®-Physical-Layer basierten Platinen erstellt (Abschn. 6.3.1). Die vorhandene Software aus dem Standard-Ethernet-Demonstrator wurde um die nötigen Funktionen erweitert. Der Signalfluss ist ähnlich wie beim ersten Demonstrator (Abb. 6.1).

Anschließend wurden standardmäßige Untersuchungen (EMV, Signalqualität) für Technologien im Fahrzeug durchgeführt. Die Ergebnisse aus den Messungen haben gezeigt, dass die Ethernet BroadR-Reach®-Technologie die EMV- und Kabellängen-Anforderungen für den Einsatz im Kfz erfüllt. Somit konnte die automobiler Tauglichkeit der Ethernet-Technologie bestätigt werden.



## 7. Zusammenfassung

Anregungen für weiterführende Arbeiten:

### Austausch der Transceiver

Auf die vorhandene Hardware- und Softwarestruktur könnten neuere Versionen der 100 Mbit/s BroadR-Reach® Transceiver adaptiert werden. Der vorhandene Broadcom BCM89810<sup>2</sup> Chip könnte durch seinen Nachfolger BCM89811<sup>3</sup> ausgetauscht werden. Ein weiterer Kandidat wäre der TJA1100<sup>4</sup> Transceiver der Firma NXP Semiconductors. Die beiden Transceiver bieten ein kleineres *Package* (6x6 mm), weniger Stromverbrauch, und der externe Filter und Spannungsregler (auf 1.8 V) sind schon auf dem Chip verbaut. Diese Verbesserungen ermöglichen eine kleinere Bauform der PCB-s, bessere EMV Eigenschaften, geringere Kosten (*bill of material* (BOM)) und Gewicht. *Samples* der beiden Transceiver sind verfügbar.

### Backbone Lösung mit 100 Mbit/s oder 1 Gbit/s Ethernet

Ein einfacher Backbone (geswitchtes Netzwerk) mit 100 Mbit/s oder 1 Gbit/s BroadR-Reach® Ethernet könnte aufgebaut werden. Mit Hilfe des Aufbaus könnte man die Echtzeit-Protokolle von AVB und TSN an verschiedenen Anwendungs-Szenarien simulieren und testen.

1 Gbit/s über eine verdrehte Zwei-Draht-Leitung befindet sich noch in der Entwicklungsphase. Die ersten *Samples* sollten aber noch heuer oder spätestens Ende 2016 erhältlich sein.

---

<sup>2</sup><http://www.broadcom.com/products/ethernet-communication-and-switching/physical-layer/bcm89810/>, Zugriffsdatum: 11.03.2018

<sup>3</sup><http://www.broadcom.com/products/Physical-Layer/BroadR-Reach-PHYs/BCM89811/>, Zugriffsdatum: 11.03.2018

<sup>4</sup>[https://www.nxp.com/products/analog/interfaces/in-vehicle-network/ethernet/automotive-phy-transceivers/ieee-100base-t1-compliant-automotive-ethernet-phy-transceiver:TJA1100HN?lang\\_cd=en/](https://www.nxp.com/products/analog/interfaces/in-vehicle-network/ethernet/automotive-phy-transceivers/ieee-100base-t1-compliant-automotive-ethernet-phy-transceiver:TJA1100HN?lang_cd=en/), Zugriffsdatum: 11.03.2018



## Literatur

- [1] Christian Baun. *Computernetze kompakt. 2., aktualisierte und erweiterte Auflage*. Springer Vieweg, 2013. ISBN: 978-3-642-41653-8.
- [2] Kai Borgeest. *Elektronik in der Fahrzeugtechnik. 3. Auflage*. Springer Vieweg, 2013. ISBN: 978-3-8348-2145-4.
- [3] Broadcom. *OPEN Alliance BroadR-Reach® (OABR) Physical Layer Transceiver Specification For Automotive Applications*. Version June 2014. Broadcom, 2014. URL: [http://www.ieee802.org/3/1TPCESG/public%20BroadR\\_Reach\\_Automotive\\_Spec\\_V3.2\\_w\\_o.pdf/](http://www.ieee802.org/3/1TPCESG/public%20BroadR_Reach_Automotive_Spec_V3.2_w_o.pdf/), Zugriffsdatum/15.04.2015.
- [4] Dr. Wilfried Steiner. *Von AVB zu TSN - Echtzeitanwendungen mit Automotive Ethernet*. Version September 2014. TTTech, 2014. URL: [https://www.tttech.com/fileadmin/content/automotive/files/pdf/Echtzeitanwendungen\\_mit\\_Automotive\\_Ethernet\\_2014-08-09.pdf/](https://www.tttech.com/fileadmin/content/automotive/files/pdf/Echtzeitanwendungen_mit_Automotive_Ethernet_2014-08-09.pdf/), Zugriffsdatum/11.03.2018.
- [5] EtherCAT Technology Group. *Industrial Ethernet Technologies: Overview*. White Paper. EtherCAT Technology Group, 2014. URL: [https://www.ethercat.org/download/documents/Industrial\\_Ethernet\\_Technologies.pdf/](https://www.ethercat.org/download/documents/Industrial_Ethernet_Technologies.pdf/), Zugriffsdatum/11.03.2018.
- [6] ETHERNET POWERLINK STANDARDIZATION GROUP. *Industrial Ethernet Facts*. White Paper. EPSG, 2011.
- [7] K. Etschberger. *Controller Area Network. 3. Auflage*. Hanser Verlag, 2006.
- [8] Feldbusse.de. *Industrial Ethernet*. Feldbusse.de. 2013. URL: <http://www.feldbusse.de/ethernetindex.asp/>, Zugriffsdatum/11.03.2018.
- [9] A. Grzemba. *LIN-Bus*. Franzis Verlag, 2005.
- [10] A. Grzemba. *MOST – The Automotive Multimedia Network. 2. Auflage*. Franzis Verlag, 2011. URL: [www.mostcooperation.com/](http://www.mostcooperation.com/), Zugriffsdatum/11.03.2018.
- [11] A. Leon-Garcia und I. Widjaja. *Communication Networks, Fundamental Concepts and Key Architectures*. McGraw Hill, 2004.
- [12] IEEE. *CFI1 Twisted Pair 100 Mbit/s Ethernet*. Version March 2014. IEEE, 2014. URL: [http://www.ieee802.org/3/cfi/0314\\_2/CFI\\_02\\_0314.pdf/](http://www.ieee802.org/3/cfi/0314_2/CFI_02_0314.pdf/), Zugriffsdatum/11.03.2018.
- [13] IEEE. *IEEE Std. 802.3*. 2005.
- [14] Texas Instruments. *HALCoGen Ethernet Driver and lwIP Integration Demonstration*. Texas Instruments. URL: [http://processors.wiki.ti.com/index.php/HALCoGen\\_Ethernet\\_Driver\\_and\\_lwIP\\_Integration\\_Demonstration/](http://processors.wiki.ti.com/index.php/HALCoGen_Ethernet_Driver_and_lwIP_Integration_Demonstration/), Zugriffsdatum/11.03.2018.

## Literatur

- [15] Texas Instruments. *Technical Reference Manual. TMS570LS31x/21x 16/32-Bit RISC Flash Microcontroller*. SPNU499B. Texas Instruments. 2013. URL: <http://www.ti.com/lit/ug/spnu499b/spnu499b.pdf/>, Zugriffsdatum/15.04.2015.
- [16] Texas Instruments. *TMS570LS31x Hercules Development Kit (HDK)- User's Guide*. SPNU509B. Texas Instruments. 2011. URL: <http://www.ti.com/lit/ug/spnu509b/spnu509b.pdf/%20>, Zugriffsdatum/11.03.2018.
- [17] Ixia. *Automotive Ethernet: An Overview*. White Paper. Version May 2014. Ixia, 2014. URL: [http://www.ixiacom.com/sites/default/files/resources/whitepaper/ixia-automotive-ethernet-primer-whitepaper\\_1.pdf/](http://www.ixiacom.com/sites/default/files/resources/whitepaper/ixia-automotive-ethernet-primer-whitepaper_1.pdf/), Zugriffsdatum/15.04.2015.
- [18] Jörg Wollert. *Echtzeit-Ethernet*. White Paper. elektroniknet.de, 2015. URL: <http://www.elektroniknet.de/messen-testen/pc-messtechnik/artikel/116241/>, Zugriffsdatum/11.03.2018.
- [19] Josef Nöbauer. *Is Ethernet the rising star for in-vehicle networks?* Continental, 2011. URL: [http://www.iestcfa.org/presentations/etfa2011/Noebauer\\_Ethernet.pdf/](http://www.iestcfa.org/presentations/etfa2011/Noebauer_Ethernet.pdf/), Zugriffsdatum/15.04.2015.
- [20] Kristof Obermann und Martin Horneffer. *Datennetztechnologien für Next Generation Networks. Ethernet, IP, MPLS und andere*. Vieweg+Teubner Verlag, 2009. ISBN: 978-3-8348-0449-5.
- [21] Mehmet Tazebay, Sessa Panguluri, Ali Abaye. *From OPEN Alliance BroadR-Reach® PHYs to IEEE 802.3bp RTPGE*. Version September 2013. Broadcom, 2013. URL: [http://standards.ieee.org/events/automotive/03\\_Tazabay\\_Broadcom\\_RTPGE.pdf/](http://standards.ieee.org/events/automotive/03_Tazabay_Broadcom_RTPGE.pdf/), Zugriffsdatum/11.03.2018.
- [22] LIN Local Interconnected Network. *LIN Specification Package Version 2.1*. LIN Local Interconnected Network, 2006. URL: [www.lin-subbus.org/](http://www.lin-subbus.org/), Zugriffsdatum/15.04.2015.
- [23] Harald Orlamünder. *Paket-basierte Kommunikationsprotokolle*. Hüthig, 2005.
- [24] Phoenix Contact. *Industrial Wireless: Drahtlos vom Sensor bis ins Netzwerk*. White Paper. Phoenix Contact, 2014. URL: [https://www.phoenixcontact.com/assets/downloads\\_ed/global/web\\_dwl\\_promotion/52000987\\_DE\\_DE\\_Industrial\\_Wireless\\_LoRes.pdf/](https://www.phoenixcontact.com/assets/downloads_ed/global/web_dwl_promotion/52000987_DE_DE_Industrial_Wireless_LoRes.pdf/), Zugriffsdatum/11.03.2018.
- [25] Werner Zimmermann und Ralf Schmidgall. *Bussysteme in der Fahrzeugtechnik*. 5. Auflage. Springer Vieweg, 2014. ISBN: 978-3-658-02419-2.
- [26] M. Rausch. *FlexRay. Grundlagen, Funktionsweise, Anwendung*. Hanser Verlag, 2007.
- [27] Konrad Reif. *Automobilelektronik*. 4. Auflage. Vieweg+Teubner Verlag, 2011. ISBN: 978-3-8348-1498-2.
- [28] Konrad Reif. *Bosch Autoelektrik und Autoelektronik*. 6. Auflage. Vieweg+Teubner Verlag, 2010. ISBN: 978-3-8348-1274-2.

- [29] Olaf Schmid. *BroadR-Reach Grundlagen*. URL: <https://olaf73.wordpress.com/2012/02/21/broadr-reach-grundlagen/>, Zugriffsdatum/11.03.2018.
- [30] Open Allience SIG. *Open Allience SIG Overview*. Open Allience SIG. 2014. URL: <http://www.opensig.org/>, Zugriffsdatum/11.03.2018.
- [31] FlexRay Communications System. *Protocol Specification Version 2.0*. LIN Local Interconnected Network, 2004. URL: [www.flexray.com/](http://www.flexray.com/), Zugriffsdatum/15.04.2015.
- [32] Andrew W. Tannenbaum. *Computernetzwerke*. Prentice Hall, 1998.
- [33] Uwe Weissenrieder. *TCP/IP in Embedded-Systemen*. White Paper. IXXAT Automation GmbH. URL: <http://www.ixxat.com/de/>, Zugriffsdatum/11.03.2018.
- [34] Vector. *MICROSAR ETH - The Vector Embedded Ethernet Stack and its Use-Cases*. Version November 2013. Vector, 2013. URL: [http://www.vector.com/portal/medien/cmc/events/Webinars/2013/Vector\\_Webinar\\_MICROSAR-ETH\\_20131211\\_DE.pdf/](http://www.vector.com/portal/medien/cmc/events/Webinars/2013/Vector_Webinar_MICROSAR-ETH_20131211_DE.pdf/), Zugriffsdatum/11.03.2018.



# Anhang



## Anhang A.

### Anhang

Im Anhang befindet sich die *Materialliste der Platine* (Abb. [A.1](#)).

# Artikel	Hersteller	Teilenummer	Beschreibung	Menge	Bezeichnung auf der Platine
1	Konnektor	Phoenix Contact	1985807	Feste Anschlussblöcke MKDS 1/2-3-5	2 CON
2	TVS-Dioden	NXP Semiconductors	PESD3V3X1BL	TVS-Dioden - Suppressorioden, 1,3pF	4 TVS1,TVS2
3	Diode	Kingbright	APT1608SGC	Standard-LEDs - SMD GREEN WATER CLEAR	4 10M_LED, 100M_LED
4	Kondensator	Taiyo Yuden	JMK325BJ226MMI-T	KONDENSATOR, 22uF, 6.3V, X5R, 1210	2 C32
5	Kondensator	Kemet	C040C105K9PACTU	1uF, 6.3V, X5R, 0603	4 C11, C17 unten
6	Tantal-kondensator	AVX	TACR106K016XTA	KONDENSATOR, 0805, 10 uF, 16V	4 C37,C38
7	Tantal-kondensator	AVX	TAJR105K016RNJ	KONDENSATOR, BAUF, R 0805, 1 uF, 16V	4 C42,C43
8	Spannungsregler	Texas Instruments	LM1085IS-3.3NOPB	SPANNUNGSREGLER, LINEAR, 3.3V	2 U1
9	Spannungsregler	Microchip	MCP1703A-1202EDB	LDO, 1.6VIN, 0.25A, 1.2V, 2%, 350T223	2 U2
10	Inductor	Würth Elektronik	744870221	INDUCTOR, DD 1280, 220uH 20%, 0.9A	2 IND
11	Kondensator	TDK	C3216X5R0J336M	KONDENSATOR, 33uF, 6.3V, X5R, 1206	2 C35
12	Kondensator	Kemet	C0805C106K8PACTU	KONDENSATOR, 0805, 10uF, 10V, X5R	6 C21,C18,C23
13	Kondensator	MURATA	GRM2165C1H330JZ01D	KONDENSATOR, 0805, 33pF, 50V, NPO	14 C1,C2,C3,C4,C5,C6,C27
14	Kondensator	MURATA	GRM2165C1H101JA01D	KONDENSATOR, 0805, 100pF, 50V, NPO, 5%	8 C39,C40,C41,C9
15	Kondensator	MULTICOMP	MC0805N180J500CT	KONDENSATOR, MLCC, 0805, 50V, 18pF	4 C7,C8
16	Kondensator	TDK	CGA4J3X7T2E104M125AE	KONDENSATOR, 0.1uF 0805, 250V +/-20%, 100nF	4 C46,C47
17	Kondensator	Multicomp	MCCA001058	KONDENSATOR, 0805, NPO, 25V, 68pF	8 C34,C36,C45,C44
18	Kondensator	Multicomp	MCCA001078	KONDENSATOR, 0805, 1%, NPO, 50V, 100pF	4 C26,C33 unten
19	Inductor, Ferrite	Murata	BLM18FG471SN1D	FERRITE PERLE, 0603 BAUFORM, 4700OHM	12 L1,L2,L3,L6,L7,L8
20	Inductor, Ferrite	Laird Technologies	MI1206K601R-10	PERLE 3216, 600/100MHz, 1500mA	4 IND1,IND2
21	Inductor	Coilcraft	0603HP-R22XGLU	HF-INDUKTIVIT 220nH, 2%, 1.1GHz SMD	8 L10,L11,L12,L13
22	Widerstand	MULTICOMP	MC01W080554K7	WIDERSTAND, 0805, 5%, 4K70	10 R8,R30,R31,R32,R34
23	Widerstand	MULTICOMP	MC01W0805533R	WIDERSTAND, 0805, 5%, 33R0	24 R16,R11,R12,R13,R14,R15,R1,R2,R3,R4,R5,R6
24	Widerstand	MULTICOMP	MC01W08055680R	WIDERSTAND, 0805, 5%, 680R	2 R33
25	Widerstand	Vishay Draloric	CRCW08051K24FKEA	WIDERSTAND, 0805, 1K24 1%, 0.125W	2 R20 unten
26	Widerstand	MULTICOMP	MC01W08055330R	WIDERSTAND, 0805, 5%, 330R	6 R28,R29,R7
27	Widerstand	MULTICOMP	MC01W08051100R	WIDERSTAND, 0805, 100R, 1%	2 R9 , unten
28	Widerstand	MULTICOMP	MCMR08X45R3FTL	WIDERSTAND, 0805 45R3, 1%	8 R25,R26,R35,R36
29	Diode	Kingbright	KPG-1608SYKC-T	LED SMD, 0603, GELB	2 ACTIVITY_LED
30	Widerstand	Vishay Draloric	CRCW080543R2FKEA	33R, 1%, 0805	12 R1,R2,R3,R4,R5,R6
31	Widerstand	Vishay Draloric	CRCW080528R0FKEA	WIDERSTAND, 0805, 28R0, 1%	2 R17
32	Widerstand	TE Connectivity	CPF0603F4R02C1	WIDERSTAND, 0603, 4R02 1% 50PPM	8 R24,R27,R10,R18
33	Widerstand	MULTICOMP	MCMR08X18R2FTL	WIDERSTAND, 18R2 , 0805	4 R19,R22
34	Widerstand	MULTICOMP	MCMR08X4990FTL	WIDERSTAND, 0805, 499R, 1%	4 R21,R23 , unten
35	Widerstand	Panasonic	ERJP06F1001TV	WIDERSTAND, SURGE, 0805, 1%, 1K	4 R37,R38
36	Crystal	TXC	7M-25.000MAAJ-T	XTAL, 25.000MHZ, 18PF, SMD, 3.2X2.5	2 CR
37	Kondensator	TAIYO YUDEN	JMK212BJ105KK-T	CAPACITOR, 1uF, 6.3V, X5R, 10%, 0805	30 C10,C13,C14,C15,C12,C22,C24,C25,C19,C20,C16,C30,C29,C28,C31

Abbildung A.1.: Materialliste der Platine