Automotive Ethernet

Learn about the latest developments in Automotive Ethernet technology and implementation with this fully revised second edition. With approximately 25% new material and greater technical detail, coverage is expanded to include

- Detailed explanations of how the 100BASE-T1 PHY and 1000BASE-T1 PHY technologies actually work
- A step-by-step description of how the 1000BASE-T1 channel was derived
- A summary of the content and uses of the new TSN standards
- A framework for security in Automotive Ethernet
- Discussion of the interrelation between power supply and Automotive Ethernet communication

Industry pioneers share the technical and nontechnical decisions that have led to the success of Automotive Ethernet, covering everything from electromagnetic requirements and physical layer technologies to Quality of Service, the use of VLANs, IP, service discovery, network architecture, and testing. This is the guide for engineers, technical managers, and researchers designing components for in-car electronics and for those interested in the strategy of introducing a new technology.

Kirsten Matheus is a communications engineer who is currently responsible for the strategy of in-vehicle networking at BMW. She has established Ethernet-based communication as an in-vehicle networking technology at BMW and within the automotive industry. She has previously worked for Volkswagen, NXP, and Ericsson.

Thomas Königseder is a communications engineer who manages the team for electromagnetic compatibility at BMW. He was responsible for the first ever Ethernet connection in a series production car with start of production in 2008.

www.cargeek.ir

Automotive Ethernet

KIRSTEN MATHEUS
BMW Munich

THOMAS KÖNIGSEDER
BMW Munich

CAMBRIDGE
UNIVERSITY PRESS

CAMBRIDGE
UNIVERSITY PRESS

University Printing House, Cambridge CB2 8BS, United Kingdom

One Liberty Plaza, 20th Floor, New York, NY 10006, USA

477 Williamstown Road, Port Melbourne, VIC 3207, Australia

4843/24, 2nd Floor, Ansari Road, Daryaganj, Delhi - 110002, India

79 Anson Road, #06-04/06, Singapore 079906

Cambridge University Press is part of the University of Cambridge.

It furthers the University's mission by disseminating knowledge in the pursuit of education, learning and research at the highest international levels of excellence.

www.cambridge.org
Information on this title: www.cambridge.org/9781107183223
DOI: 10.1017/9781316869543

First published 2015
Second edition 2017

Printed in the United Kingdom by TJ International Ltd. Padstow Cornwall

A catalogue record for this publication is available from the British Library

Library of Congress Cataloging-in-Publication data
Names: Matheus, Kirsten, author. | Königseder, Thomas, author.
Title: Automotive ethernet / Kirsten Matheus, BMW Munich, Thomas Königseder, BMW Munich.
Description: Cambridge, United Kingdom ; New York, NY, USA : Cambridge University Press, [2017] | Includes bibliographical references and index.
Identifiers: LCCN 2016059791 | ISBN 9781107183223 (hardback : alk. paper)
Subjects: LCSH: Automobiles–Computer networks. | Ethernet (Local area network system)
Classification: LCC TL272.53 .M38 2017 | DDC 629.2/72–dc23 LC record available at https://lccn.loc.gov/2016059791

ISBN 978-1-107-18322-3 Hardback

Contents

www.cargeek.ir

Preface to the First Edition

On 11 November 2013, I, Kirsten Matheus, attended a celebration of 40 years since the invention of Ethernet at an IEEE 802 plenary meeting. During the celebration, Robert Metcalfe, David Boggs, Ronald Crane, and Geoff Thompson were honored as the pioneers of Ethernet. If I had to name the people without whom Automotive Ethernet would not have happened as it did, I would name Thomas Königseder, technical expert at BMW and coauthor of this book, and Neven Pischl, EMC expert at Broadcom.

It all started in 2004, when Thomas received the responsibility for speeding up the software flash process for BMW cars. With the CAN interface used at the time, flashing the 1 Gbyte of data anticipated for 2008 would have required 16 hours to complete. After careful evaluation, Thomas chose and enabled the use of standard 100Base-TX Ethernet for this purpose. Thus, in 2008, the first serial car with an Ethernet interface, a BMW 7-series, was introduced to the world.

This was only the beginning though. The problem was that the EMC properties of standard 100Base-TX Ethernet were not good. So the technology was usable with cost-competitive unshielded cables only when the car was stationary in the garage for the specific flash use case. To use 100Base-TX also during the runtime of the car would have required shielding the cables, and that was too expensive.

Yet, Thomas was taken with the effectiveness of Ethernet-based communication and therefore investigated ways to use 100Base-TX over unshielded cables. He identified the problem but could not solve it. So, in 2007, he contacted various well-established Ethernet semiconductor suppliers to work with him on a solution. Only Broadcom responded positively, and engineers from both companies evaluated the BMW 100Base-TX Ethernet EMC measurements. Then, in January 2008, it happened: BMW performed EMC measurements with boards the Broadcom engineer Neven Pischl had optimized using a 100 Mbps Ethernet PHY variant Broadcom had originally developed for Ethernet in the First Mile and which Broadcom engineers had further adapted for the automotive application. The very first measurements ever performed at a car manufacturer with this technology were well below the limit lines and yielded better EMC performance results than even the existing FlexRay!

This was when Automotive Ethernet was born. Without having had this technology available at the right time, without proving that 100 Mbps can be transmitted over Unshielded Twisted Pair (UTP) cables in the harsh automotive EMC environment, none of all the other exciting, complementary, futuristic, and otherwise useful developments in the field would have happened. BMW would likely be using Media Oriented Systems

Transport (MOST) 150 and be working on the next speed grade of MOST together with the rest of the industry.

Naturally, from the discovery of a solution in 2008 to the first ever introduction of the UTP Ethernet in a serial car, a BMW X5, in 2013, and, to establishing Automotive Ethernet in the industry was and is a long run. Thomas and I would therefore like to thank all those who helped to make this happen and those who are today fervently preparing the bright future Ethernet has in automotive, inside and outside of BMW, with a special mention of Stefan Singer (Freescale), who, among other things, established the first contact between BMW and Broadcom. Using Ethernet for in-car networking is a revolution, and it is an unparalleled experience to be able to participate in its development.

This book explains the history of Automotive Ethernet in more detail and also how Automotive Ethernet can technically be realized. We would like to thank all those who supported us with know-how and feedback in the process of writing this book. First, we thank Thilo Streichert (Daimler), who made it his task to review it all and who saved the readers from some of the blindness that occurs to authors having worked on a particular section for too long. Then there are (in alphabetical order) Christoph Arndt (FH Deggendorf), Jürgen Bos (Ericsson, EPO), Karl Budweiser (TU München), Steve Carlson (HSPdesign), Bob Grow (RMG Consulting), Mickael Guilcher (BMW), Robert von Häfen (BMW), Florian Hartwich (BOSCH), Thomas Hogenmüller (BOSCH), Michael Johas Teener (Broadcom), Michael Kaindl (BMW), Oliver Kalweit (BMW), Ramona Kerscher (FH Deggendorf), Matthias Kessler (ESR Labs), Max Kicherer (BMW), Yong Kim (Broadcom), Rick Kreifeld (Harman), Thomas Lindenkreuz (BOSCH), Thomas Lindner (BMW), Stefan Schneele (EADS), Mehmet Tazebay (Broadcom), Lars Völker (BMW), Ludwig Winkel (Siemens), and Helge Zinner (Continental). Last, but not least, we would like to thank BMW for supporting our work for the book.

Preface to the Second Edition

In September 2011, Automotive Ethernet was still at its very beginning. BMW was far and wide the only car manufacturer seriously interested. In 2011, BMW had been in production with 100BASE-TX for diagnostics and flash updates for three years and had decided to go into production in 2013 with what is now called 100BASE-T1 in its new surround view system.

In September 2011, strong doubts still had the upper hand. The main concern was that transmitting Ethernet packets at 100 Mbps over a single Unshielded Twisted Pair (UTP) cable would not be possible under the harsh automotive electromagnetic conditions. Another concern was the missing ecosystem. At the time, there was only one supplier of the transceiver technology, Broadcom, which had no prior experience with the written and unwritten requirements of the automotive industry. Additionally, BMW was only just starting to involve the supporting industry of test institutions, tool vendors, software houses, and so on.

For an outsider, September 2011 was thus a time of uncertainty. From the inside, however, it was a time in which the foundations for the success of Automotive Ethernet were being laid and during which we ensured that the right structural support was in place. In the background, we were finalizing the framework of the OPEN Alliance; NXP was in full speed, evaluating its chances as a second transceiver supplier; and BMW was preparing to congregate the industry at the first Ethernet&IP@Automotive Technology Day; while first discussions on starting the next-generation standardization project, 1000BASE-T1, concurred.

One of my, Kirsten Matheus's, many jobs at the time was to interest more semiconductor vendors in Automotive Ethernet. In September 2011, this meant getting them to negotiate a licensing agreement with Broadcom, one of their competitors, while the market prospects were still foggy. In one of the discussions I had, an executive manager explained to me in detail why this was out of question, based on the following experience.

In the past, he had worked for another semiconductor company that was addressed by a powerful customer to be the second supplier for a proprietary Ethernet version (just like 100BASE-T1 was proprietary in September 2011, when itwas still BroadR-Reach and neither published in the OPEN Alliance or by IEEE). This customer of theirs offered significantly higher volumes than BMW ever could, and it was even in the position to technically support them with interoperability and other technical questions, which the executive manager did not expect BMW to be capable of. They invested in and developed a respective Ethernet PHY product.

However, shortly after, the IEEE released an Ethernet specification for the same use case. This IEEE version was seen as technologically inferior. However, it had one technical advantage over the proprietary technology the executive manager's company had invested in: It was backward compatible to previously designed IEEE Ethernet technologies. The IEEE technology prevailed, whereas the solution they had invested in never gained any serious traction. In consequence, they would not again invest in a technology that was not a public standard. The prospect of the OPEN Alliance acting as an organization ensuring transparency with respect to licensing and technical questions did not make any difference to them.

Today, five years later, in 2016, we know that if that semiconductor company had invested in 100BASE-T1/BroadR-Reach in 2011, its business prospects today would be excellent – not only because the technology persevered but also because the company would have been early in the market. Was the executive all wrong in saying that it needs to be a public standard? I do not know.

Many things happened in the meantime. Based on experiences with BroadR-Reach/100BASE-T1, what BMW had wanted to begin with became doable: transmitting 100 Mbps Ethernet over unshielded cables during runtime using 100BASE-TX PHYs. This solution, sometimes called Fast Ethernet for Automotive (FEFA), was based on a public IEEE standard. For BMW, it came too late. But most other car manufacturers had not yet made any decisions. For a while, it was not certain whether the "proprietary" (but licensed) BroadR-Reach would succeed in the market or the tweaked "public" 100BASE-TX.

Well, today we know: BroadR-Reach made it. But, in the meantime, it has also become a public standard, called IEEE 802.3bw or 100BASE-T1. Only three weeks after handing in the manuscript of the first edition for this book, a respective Call For Interest (CFI) successfully passed at IEEE 802.3. The IEEE released a "BroadR-Reach compliant" specification as an IEEE 802.3 standard in October 2015. Maybe BroadR-Reach would have succeeded even without IEEE's blessing. Who knows? The fact is, the IEEE standardization made life easier. It erased the topic of technology ownership from discussion.

And it was a main motivator to write this second edition. The now publicly available 100BASE-T1 and BroadR-Reach specifications allowed us to go into detail. The reader will thus find a significantly extended PHY chapter, which now includes a detailed explanation of the 100BASE-T1 and 1000BASE-T1 technologies, whose standardization has also been completed in the meantime. While the description of the 100BASE-T1 technology includes experiences while implementing and using the technology, the 1000BASE-T1 description includes the methodology used behind developing a technology in case of an unknown channel – something new and useful also for future development projects.

Furthermore, the PHY chapter now has a distinct power supply section. Specifications on wake-up and Power over Dataline (PoDL) been released in the meantime, and are in need of context. Additionally, power supply impacts the EMC behavior. How this influences Automotive Ethernet is also described. On the protocol layer, there are new developments in respect to Time-Sensitive Networking, discussions of which have been included in the protocol chapter. Furthermore, the security section has been extended significantly. Last, but not least, we have updated all chapters with the latest developments and insights.

Like the first edition, this edition would not have been possible without the support of the colleagues who make Automotive Ethernet happen on a daily basis. For this second edition, we would especially like to extend our gratitude to (in alphabetical order)

- Karl Budweiser, BMW, who had the (mis)fortune to start working at BMW just at the right time to proofread the PHY section
- Thomas Hogenmüller, BOSCH, who did not contribute directly to this book but who successfully dared to drive the standardization of BroadR-Reach at IEEE, and without whom the main reason for writing this second edition might not have happened
- Thomas Lindner, BMW, who dissected the BroadR-Reach/100BASE-T1 technology and was thus able to contribute vital insights to the 100BASE-T1 description – the reader will benefit greatly from his scrutiny
- Brett McClellan, Marvell, who answered many questions on the 1000BASE-T1 specification and helped in understanding the technology
- Mehmet Tazebay, Broadcom, who, as the key designer of BroadR-Reach/100BASE-T1 and 1000BASE-T1, has not only provided the basis for what happened in Automotive Ethernet as such but also answered many questions
- Michael Ziehensack, Elektrobit, whose insights supported the security section
- Helge Zinner, Continental, who relentlessly counterread the complete second edition and made it a significantly more consistent and precise book than it would have been without him

Last, but not least, we would like to thank BMW for supporting our work on the book and for giving us the opportunity to make a difference.

Abbreviations

#	number of
1PPoDL	One Pair Power over Data Line
2D	Two-Dimensional
3B2T	Three Bits to Two Ternary conversion
3D	Three-Dimensional
4B3B	Four Bits to Three Bits conversion
4D	Four-Dimensional
AAA2C	Avnu sponsored Automotive Avb gen 2 Council
AAF	AVTP Audio Format
AAN	Automotive Area Network
ACK	Acknowledgment
ACL	Access Control List
ACR-N	Attenuation to Cross talk Ratio at Near end
ACR-F	Attenuation to Cross talk Ratio at Far end
ADAS	Advanced Driver Assist System
ADC or A/D	Analog to Digital Converter
ADSL	Asynchronous Digital Subscriber Line
AEC	Automotive Electronics Council
AFDX	Avionics Full-Duplex Switched Ethernet
AFEXT	Alien Far-End Cross Talk
AGC	Adaptive Gain Control
AIDA	AutomatisierungsInitiative der Deutschen Automobilhersteller (English: Automation Initiative of German Automobile manufacturers)
ALOHA	Hawaiian greeting, name for the multiple user access method developed at the University of Hawaii
AM	Amplitude Modulation
AMIC	Automotive Multimedia Interface Corporation
Amp. or AMP	Amplifier
ANEXT	Alien Near-End Cross Talk
ANSI	American National Standards Institute
API	Application Programming Interface
APIX	Automotive PIXel link

ARINC	Aeronautical Radio Inc., a company founded in 1929, which today, among other things, publishes communication standards for the aerospace/aviation industry
ARP	Address Resolution Protocol
ARPANET	Advanced Research Projects Agency NETwork
ASIC	Application Specific Integrated Circuit
ASN	Avionics Systems Network
ATM	Asynchronous Transfer Mode, a telecommunications protocol used in networking
AUTOSAR	AUTomotive Open System ARchitecture, organization for the development of standards in for software development in automotive
AV, A/V	Audio Video
AVB	Audio Video Bridging, refers to a set of IEEE standards
AVBgen1	First generation of IEEE AVB standards
AVBgen2	Second generation of IEEE AVB standards, renamed TSN
AVnu	Includes the AV for Audio Video and also means "road" in Creole [1]
AVS	Audio Video Source
AVTP	AVb Transport Protocol based on IEEE 1722
AWGN	Additive White Gaussian Noise
AXE	Name of Ericsson's digital switch product line
B	Billion
BAG	Bandwidth Allocation Gap
BCI	Bulk Current Injection
BLW	BaseLine Wonder correction
BM	Bus Minus (FlexRay terminology)
BMCA	Best Master Clock selection Algorithm
BP	Bus Plus (FlexRay terminology)
BPDU	Bridge Protocol Data Units
BSD	Berkeley Standard Distribution or Berkeley Software Distribution
C2C	Car-to-Car (communication)
C2X	Car-to-anything (communication)
CAGR	Compound Annual Growth Rate; CAGR = $(\text{Volume}_{t2}/\text{Volume}_{t1})^{(1/(t2-t1))} - 1$
CAN	Controller Area Network
CAN FD	CAN with Flexible Data rate
CC	Communication Controller
CCITT	Comité Consultatif International Téléphonique et Télégraphique, renamed ITU-T in 1993
CD	Compact Disc
CDM	Charged Device Model
CE	Consumer Electronics or Carrier Ethernet
CFI	Call For Interest
CIDR	Classless Inter-Domain Routing

CISPR	Comité International Spécial des Perturbations Radioélectriques
CM	Common Mode
CMC	Common Mode Choke
CML	Current Mode Logic
COTS	Commercial-Off-The-Shelf
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check, a form of channel coding used to detect (and sometimes correct) errors in a transmission
CRF	Clock Reference Format
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
CSN	Coordinated Shared Network
D^2B	Domestic Digital Bus
DAC or D/A	Digital to Analog Converter
DAS	Driver Assist Systems or Driver ASsist
DC	Direct Current
DEI	Drop-Eligible Identifier
DFE	Decision Feedback Equalizer
DHCP	Dynamic Host Configuration Protocol
DIX	Dec Intel Xerox
DLNA	Digital Living Network Alliance
DLL	Data Link Layer
DM	Differential Mode
DMA	Direct Memory Access
DMLT	Distinguished Minimum Latency Traffic
DNS	Domain Name System
DPI	Direct Power Injection
DRM	Digital Rights Management
DSP	Digital Signal Processor
DSQ 128	Double Square constellation, 2 times 16 discrete levels of PAM16 mapped on a two-dimensional checkerboard
DUT	Device Under Test
EADS	European Aeronautic Defence and Space Company, Airbus is a division of EADS
EAP	Ethernet Authentication Protocol
ECU	Electronic Control Unit
EE or E/E	Electric Electronic
EEE	Energy-Efficient Ethernet
EFM	Ethernet in the First Mile (IEEE 802.3ah)
EIA	Electronic Industries Alliance
ELFR	Early Life Failure Rate
ELTCTL	Equal Level Transverse Conversion Transfer Loss
EMC	ElectroMagnetic Compatibility
EMD	Electronic Master Device

EME	ElectroMagnetic Emissions
EMI	ElectroMagnetic Immunity (in other document sometimes also used for ElectroMagnetic Interference, which can mean the opposite)
EMS	Electro Magnetic Susceptibility, more common: EMC immunity
EPO	European Patent Office
EPON	Ethernet Passive Optical Network (part of EFM)
ESD	ElectroStatic Discharge or End Stream Delimiter
Eth.	Ethernet
Euro NCAP	European New Car Assessment Programme
EWSD	Elektronisches WählSystem Digital (English: Electronic Digital Switching System/Electronic World Switch Digital)
FBAS	FarbBildAustastSynchron signal (English: CVBS, Color, Video, Blanking, and Synchronous signal)
FCC	Federal Communications Commssion
FCDM	Field induced Charge Device Model
FCS	Frame Check Sequence
FEC	Forward Error Correction
FEFA	Fast Ethernet For Automotive
FFE	Feed Forward Equalizer
FIFO	First In First Out
FlexRay	A serial, deterministic, and fault-tolerant fieldbus for automotive use
FOT	Fiber Optical Transmitter
FPD	Flat Panel Display
fps	frames per second
FRAND	Fair, Reasonable And Non-Discriminatory (the European equivalent of RAND)
FTZ	Forschungs- und Transfer Zentrum (english: research and transfer center), part of the University of Applied Science in Zwickau, Germany
GB	Giga Bytes
Gbps	Giga bit per second
GENIVI	Is a word construct taken from Geneva, the international city of peace, in which apparently the concept of GENIVI was publically presented for the first time, and In-Vehicle Infotainment [2]
GDP	Gross Domestic Product
GEPOF	Gigabit Ethernet over Plastic Optical Fiber (1000BASE-RH, IEEE802.3bv)
GMII	Gigabit Media-Independent Interface
GND	GrouND
GOF	Glass Optical Fiber
GPS	Global Positioning System
gPTP	generalized Precision Time Protocol
h	hour
H.264	also MPEG-4 Part 10 or Advanced Video Coding, video compression standard of ITU-T

HBM	Human Body Model
HD	High Definition
HDCP	High-bandwidth Digital Content Protection
HDMI	High-Definition Multimedia Interface
HE	High End
HF	High Frequency
hi-fi	High Fidelity, term used to refer to high-quality reproduction of sound in the home, invented in 1927 [3]
HMI	Human Machine Interface
HPF	High Pass Filter
Hres	Horizontal RESolution
HS CAN	High-Speed CAN
HSE	High-Speed Ethernet, Industrial Ethernet variant of the Fieldbus Foundation
HSFZ	High-Speed Fahrzeug Zugang (English: High-Speed Car Access), BMW term for the vehicle diagnostic/flash interface using Ethernet
HSM	Hardware Security Module
HTTP	HyperText Transfer Protocol (loads website into a browser)
HU	Head Unit, main infotainment unit inside the car (former radio)
I^2C	Inter-Integrated Circuit, referred to also as I-two-C or IIC
I^2S	Inter-IC Sound, Integrated Interchip Sound, or IIS
IANA	Internet Assigned Numbers Authority
IC	Integrated Circuit
ICMP	Internet Control Message Protocol
ID	IDentifier, IDentification
IDL	Interface Definition Language or Interface Description Language
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IEEE-RA	IEEE Registration Authority
IEEE-SA	IEEE Standards Association
IET	Interspersing Express Traffic
IETF	Internet Engineering Task Force
IFE	In-Flight Entertainment
IGMP	Internet Group Management Protocol
IL	Insertion Loss or attenuation
IMAP	Internet Message Application Protocol
infotainment	INFOrmation and enterTAINMENT
INIC	Intelligent Network Interface Controller
I/O	Input/Output
IP	Industrial Protocol or Internet Protocol or Intellectual Property
IPC	InterProcess Communication
IPR	Intellectual Property Rights
IPsec	Internet Protocol SECurity
ISI	InterSymbol Interference

ISO	International Organization for Standardization
IT	Information Technology
ITU-T	International Telecommunication Union – Telecommunications standardization sector
IVN	In-Vehicle Networking
JASPAR	Japan Automotive Software Platform and ARchitecture
JPEG	Joint Photographic Experts Group (standardized in ISO/IEC 10918–1, CCITT Recommendation T.81), describes different methods for picture compression
kbps	kilo bit per second
K-Line	Name for a single-ended, RS-232 similar technology standardized in ISO 9141–2
LAN	Local Area Network
LCL	Longitudinal Conversion Loss
LCTL	Longitudinal Conversion Transmission Loss
LDPC	Low Density Parity Check
LED	Light Emitting Diode
LIN	Local Interconnect Network
LLC	Logical Link Control
LLDP	Link Layer Discovery Protocol
LPF	Low Pass Filter
LPI	Low-Power Idle
LS CAN	Low-Speed CAN
LVDS	Low-Voltage Differential Signaling
MAAP	Mac Address Acquisition Protocol
MAC	Media Access Control
MB	Mega Bytes
Mbps	Mega bit per second
MEF	Metro Ethernet Forum
MDC	Management Data Clock
MDI	Media-Dependent Interface
MDIO	Management Data Input/Output
MGbps	Multi-Gigabit per second
MHL	Mobile High-definition Link
MIB	Management Information Base
MIDI	Musical Instrument Digital Interface manufacturers association
min	minutes
Mio	Millions
MJPEG	Motion JPEG
MLB	Media Local Bus (specified for MOST)
MM	Machine Model
MMRP	Multiple Mac Registration Protocol
MOST	Media Oriented Systems Transport
MOST Co	MOST Cooperation

MP3	MPEG-1 Audio Layer III (MPEG 1 Part 3) or MPEG-2 Audio Layer III (MPEG-2 Part 3)
MPEG	Moving Picture Experts Group, set standards for audio/video compression and transmission
MPEG2-TS	MPEG No. 2-Transport Stream
MPLS	Multi-Protocol Label Switching
MQS	Micro Quadlock System (type of connector)
MSE	Mean Square Error
Msps	Mega symbols per second, equals MBaud
MSRP	Multiple Stream Reservation Protocol
MVRP	Multiple VLAN Registration Protocol
μC	MicroController
n/a	not available or not applicable
NACK	Packet has not been received as expected (Negative ACK)
NAT	Network Address Translation
NC	Numerically Controlled
NEXT	Near-End Cross Talk
NIC	Network Interface Controller
NM	Network Management
nMQS	nano MQS (smaller version of the MQS connector)
NRZ	Non Return to Zero, two level signaling often used for optical transmission
ns	nanoseconds
OABR	Open Alliance Broadr-Reach sometimes also referred to as UTSP Ethernet or as simply as BroadR-Reach (now also 100BASE-T1)
OAM	Operation, Administration, Management
OBD	On-Board Diagnostic
OCF	Open Connectivity Foundation
OEM	Original Equipment Manufacturer, in the automotive industry often uses as a synonym for car manufacturer
OPEN	One Pair EtherNet alliance
OS	Operating System
OSEK	"Offene Systeme und deren Schnittstellen für die Elektronik im Kraftfahrzeug" (English: Open systems and their interfaces for electronics in automobiles)
OSI	Open Systems Interconnection
P2MP	Point-to-Multipoint (refers to a form of sharing the medium)
P2P	Point-to-Point (can, in another context, also mean Peer-to-Peer)
PAM	Pulse Amplitude Modulation
PAMx	x-level Pulse Amplitude Modulation
PAN	Personal Area Network
PC	Personal Computer
PCB	Printed Circuit Board
PCS	Physical Coding Sublayer

PD	PhotoDiode
PHY	PHYsical Layer, refers to the physical signaling and media, Layer 1 of the ISO/OSI layering model
PICS	Protocol Implementation Conformance Statement
PLC	Programmable Logic Controller or Power Line Communication
PLL	Phase-Locked Loop
PMA	Physical Medium Access
PMD	Physical Medium Dependent
PoDL	Power over DataLine, used for transmission of power over an (Ethernet) data line independent from the number of pairs needed
PoE	Power over Ethernet, note that this refers directly to the implementation described in IEEE 802.3af (which was later incorporated as clause 33 into the revision document IEEE 802.3–2005) and IEEE 802.3at focusing on the 2 pair 100Base-TX Ethernet.
POF	Polymeric/Plastic Optical Fiber
POSIX	Portable Operating System Interface
PPM	Parts Per Million, sometimes also called Defects Per Million (DPM)
PSA	Peugeot Société Anonyme
PS-ACR-F	Power Sum Attenuation to Cross talk Ratio at Far end
PS-ACR-N	Power Sum Attenuation to Cross talk Ratio at Near end
PS-NEXT	Power Sum for Near-End Cross Talk
PSAACRF	Power Sum for Alien Attenuation to Cross talk Ratio at Far end
PSANEXT	Power Sum for Alien Near-End Cross Talk
PSD	Power Spectral Density
PSTN	Public Switched Telephone Network
PTP	Precision Time Protocol
PTPv2	PTP version 2 (based on IEEE 802.1AS instead of on the older IEEE 1588, PTPv1)
QoS	Quality of Service
RAND	Reasonable And Non-Discriminatory
RARP	Reverse Address Resolution Protocol
RFC	Request for Comment
RFI	Radio Frequency Interference
RfQ	Request for Quote
RGB	Red Green Blue, analog video transmission based on transmitting one color per cable
RL	Return Loss or echo
ROM	Read Only Memory
RPC	Remote Procedure Call
RS-FEC	Reed-Solomon Forward Error Correction
RS-232	A binary, serial interface first introduced by the EIA in 1962
RSE	Rear Seat Entertainment
RTP	Real-time Transport Protocol
RTPGE	Reduced Twisted Pair Gigabit Ethernet

Rx / RxD	Receiver ingress
S-parameter	Scattering-parameter
SAE	Society of Automotive Engineers
SD	Service Discovery
SDH	Synchronous Digital Hierarchy (technology for core telecommunications networks)
SecOC	SECure On-board Communication (AUTOSAR specification)
SEIS	Sicherheit in Eingebetteten Ip-basierten Systemen (English: Security in Embedded IP-based Systems)
Semicond.	Semiconductor(s)
SER	Symbol Error Rate
SerDes	Serializer Deserializer, SerDes interfaces are named “pixel links” in this book (sometimes also called “high-speed video links” or, technically imprecise, “LVDS”)
SFD	Start Frame Delimiter
SIG	Special Interest Group
SL	StripLine
SMTP	Simple Mail Transfer Protocol
SNR	Signal-to-Noise Ratio
SOA	Service-Oriented Architecture
SoC	System on Chip
SOME/IP	Scalable service-Oriented MiddlewarE over IP
SONET	Synchronous Optical NETworking (technology for core telecommunications networks)
SOP	Start Of Production
SPI	Serial Peripheral Interface
SQI	Signal Quality Indicator
SR	Stream Reservation
SRP	Stream Reservation Protocol
SRR	Substitute Remote Request
SSD	Start Stream Delimiter
SSL	Secure Sockets Layer
SSO	Standard Setting Organization
STP	Shielded Twisted Pair or Spanning Tree Protocol
SUV	Service Utility Vehicle
SD-DVCR	Standard Definition Digital Video Cassette Recorder
SVS	Surround View System
SW	SoftWare
TAS	Time-Aware Shaping
TC	Technical Committee
TCI	Tag Control Information
TCL	Transverse Conversion Loss
TCM	Trellis Coded Modulation
TCP	Transmission Control Protocol

www.cargeek.ir

TCTL	Transverse Conversion Transfer Loss
TDM	Time Division Multiplexing, also used as a synonym for circuit-switched networks
TEM	Transversal ElectroMagnetic (wave)
TIA	Telecommunications Industry Association or TransImpedance Amplifier
TLS	Transport Layer Security
TP	Twisted Pair
TSMC	Taiwan Semiconductor Manufacturing Company
TSN	Time-Sensitive Networking
TTL	Time-To-Live
Tx / TxD	Transmitter egress
UBAT	battery voltage
UDP	User Datagram Protocol
UDS	Unified Diagnostic Services
UNFCCC	United Nations Framework Convention on Climate Change
Unix	Derived from Uniplexed Information and Computing Service (UNICS)
UPnP	Universal Plug and Play
USB	Universal Serial Bus
USP	Unique Selling Proposition, Unique Selling Point
UTP	Unshielded Twisted Pair
UTSP	Unshielded Twisted Single Pair, if combined with Ethernet, this often also refers to OABR
UWB	Ultra Wide Band, IEEE 802.15.4a
VAN	Vehicle Area Network
VCC	Pin for IC voltage supply (VCC or VDD depends on the type of IC used)
VDA	Verband der Automobilindustrie (English: German Association of the Automotive Industry)
VDD	Pin for IC voltage supply (VCC or VDD depends on the type of IC used)
VDE	Verband Deutscher Elektrotechniker (English: Association for Electrical, Electronic and Information Technologies)
VLSM	Variable Length Subnet Mask
VCIC	Video Communication Interface for Cameras
VID	Vlan IDentifier
VIN	Vehicle Identification Number
VL	Virtual Link
VLAN	Virtual LAN
VoIP	Voice over IP
Vpp	Volts peak to peak
Vres	Vertical resolution
WAN	Wide Area Network

www.cargeek.ir

WiFi	marketing name invented by the WiFi Alliance for IEEE 802.11 enabled WLAN products; WiFi is often synonymously used for WLAN [4]
WLAN	Wireless LAN
WPAN	Wireless PAN
WRAN	Wireless Regional Area Network

References

[1] R. Kreifeld, *Unpublished e-mail correspondence*, 2013.
[2] GENIVI Alliance, "GENIVI FAQ," 22 July 2013. [Online]. Available: www.genivi.org/sites/default/files/GENIVI_FAQ_072213.pdf. [Accessed 11 October 2013].
[3] Hartley Joudspeakers, "A brief history," 2013. [Online]. Available: www.hartleyloudspeakers.com/new_page_1.htm. [Accessed 30 October 2013].
[4] Wikipedia, "WiFi," 1 November 2013. [Online]. Available: http://en.wikipedia.org/wiki/Wi-Fi. [Accessed 1 November 2013].

Timeline

1965	AT&T installs the world's first electronic telephone switch (special purpose computer) in a local telephone exchange [1].
1968	Invention of Programmable Logic Controllers (PLCs) [2].
1969	AT&T employees at Bell Labs develop the operating system Unix, which eventually enabled distributed computing with remote procedure calls and the use of remote resources. For antitrust reasons, AT&T was neither allowed to sell Unix nor to keep it to itself. In consequence, they shipped it to everyone interested [3].
1969 Apr. 7	The RFC 1 is published [4]. It discusses the host software for ARPANET's switching nodes. ARPANET represents one of the world's first operational packet switching networks [5].
1969 Oct. 29	The first ARPANET link is established between University of California, Los Angeles and Stanford Research Institute [6].
1971 Nov. 3	Publication of the first "UNIX Programmer's Manual" [7].
By 1973	Unix was recoded in C (it was first developed in (an) Assembly language). This greatly enhanced Unix's portability to different hardware and further incited its distribution [7].
1973	The International Electrotechnical Commission (IEC) creates a technical committee (TC77) to specifically handle questions of electromagnetic compatibility [8].
1973 May 22	First documentation of Ethernet as an idea in a memo from Robert Metcalfe at Xerox PARC [9]. At that time, Xerox PARC was selling the first personal computer workstations (called "Xerox Alto") and had invented the first laser printers [10]. Metcalfe was working on a solution for data transmission between these products and the early Internet.
1973 Oct.	Unix was presented publicly to the Fourth Association for Computer Machinery on Operating System Principles [3].
1973 Nov. 11	First Xerox internal demonstration of Ethernet [9].
1974 Dec.	Release of the "Specification of Internet Transmission Control Protocol" (TCP), RFC 675, which was initiated by the Defense Advanced Research Projects Agency, influenced by early networking protocols from Xerox PARC and refined by the Networking Research Group of the University of Stanford [11].

1975	Honeywell and Yokogawa introduce the first distributed computer control systems for industrial automation [12].
1975 Mar. 31	Xerox files a patent application listing Robert Metcalfe, David Boggs, Charles Thacker, and Butler Lampson as inventors [13].
1976 Jul.	First paper published on Ethernet [14].
1977	The ISO formed a committee on Open System Interconnection (OSI) [15]. Somewhat later a group from Honeywell Information Systems presented their seven layer model to the ISO OSI group [16].
1978 Mar. 9	The Computer System Research Group of the University of California, Berkeley released its first Unix derivative, the Berkeley Software Distribution (BSD) [17].
1978 Apr. 1	ARINC publishes the first ARINC 429 communication standard for avionic equipment [18].
1979 Jun.	ISO publishes the OSI layering model [16].
1979 Jun. 4	Metcalfe founds 3COM to build Ethernet competitive products and convinces DEC, Intel, and Xerox (referred to as DIX) to use and promote Ethernet as a standard for their products [9] [19].
1979–82	Next to 3COM, several start-up companies were founded that built Ethernet products. The most successful ones in the mid 1980s were Ungermann-Bass (U-B), Interlan, Bridge Communications, and Excelan [19].
1980 Feb.	IEEE starts 802 project to standardize LANs [19].
1980 May	The DIX group joins the IEEE 802 project and offers Ethernet for adoption while still working on it [19].
1980 Aug. 29	The User Datagram Protocol (UDP) was published as RFC 768 [20].
1980 Sep. 30	Publication of the first version of the so-called DIX Standard (from DEC/Intel/Xerox) on Ethernet. At 2.94 Mbps, it was able to support 256 devices [21].
1980 Dec.	IEEE 802 LAN effort was split into three groups: 802.3 for CSMA/CD (Ethernet), 802.4 for Token Bus (for the factory automation vendors), and 802.5 for Token Ring (drive by IBM) [19].
1981 Mar.	3Com shipped its first 10 Mbps Ethernet 3C100 transceiver [22].
1981 Sep.	With the fourth version the Transmission Control Protocol (TCP) and the Internet Protocol are published in separate documents, RFC 793 [23] and RFC 791 [24].
1982 Aug.	Simple Mail Transfer Protocol (SMTP) is published as RFC 821 [25].
1982 Sep.	3COM ships the first Ethernet adapter for IBM PCs [9].
1982 Nov.	The second version of the DIX Ethernet Standard is published [26].
1983	IEEE publishes 802.3 10BASE-5 for 10 Mbps over thick coax cable [27].
1983	The trade press names at least 21 companies either developing or manufacturing Ethernet products: the five startups (3Com, U-B, Interlan, Bridge Communications, and Excelan), eight computer manufacturers (DEC, H-P, Data General, Siemens, Tektronix, Xerox, ICL, and NCR),

	and seven chip manufacturers (Intel, AMD, Mostek, Seeq, Fujitsu, Rockwell, and National Semiconductors), all fiercely competing [19].
1983	BOSCH starts a company-internal project to develop CAN [28].
1984 Jan. 1	AT&T monopoly is broken up; existing installed telephone wiring was usable for their services by competing companies [1].
By 1985	Approximately 30,000 Ethernet networks had been installed, connecting at least 419,000 nodes [19].
1985	IEEE publishes 802.3 10BASE-2 for 10 Mbps over thin coax cable [27].
1986	Market introduction of Token Ring, quickly gaining momentum as it was able to use the telephone wires, was more reliable, and was easier to trouble shoot [19].
1987	Two hundred vendors of Ethernet equipment counted [19].
Mid 1987	SynOptics (Xerox spinout) shipped the first (proprietary) 10 Mbps Ethernet version for telephone wire. Even if this solution was proprietary, it proved its feasibility [19].
1987 Dec.	BMW introduces the first car with a communication bus for diagnostic purposes.
1988	The all-electronic fly-by-wire system is introduced into commercial airplane service (on the Airbus A320) [29].
1989 Oct.	Publication of the TCP/IP Internet Protocol (IP) suite as "Requirements for Internet Hosts – Communication Layers," RFC 1122 [30], and "Requirements for Internet Hosts – Application and Support," RFC 1123 [31].
1989–90	The World Wide Web is invented at CERN [32].
1990 Sep.	IEEE 802.3 ratified 10BASE-T [27] (with some effort, as various proprietary solutions had evolved [19]). Ethernet had won the battle against competing technologies, by adapting to market realities and shifting from coax to twisted pair [9].
1991	TIA publishes TIA-568. It describes an inexpensive and easy to maintain UTP structured wiring plant. This includes the definition of pin/pair assignments for eight-conductor 100 Ohm balanced twisted pair cabling for wires in 8P8C/RJ-45 eight-pin modular connector plugs and sockets [33].
1992	The first cars using CAN roll off the assembly line at Mercedes Benz [28].
1993	IEEE 802.3 releases 10BASE-F, its first of a large number of optical versions [27].
1994 Jun.	Initial release of the first automotive quality specification for integrated circuits AEC-Q100 [34].
1995	The first commercial VoIP product allows real-time, full-duplex voice communication over the Internet using 1995 available hardware and bandwidth [35].
1995	IEEE 802.3 releases 100BASE-TX (-T4, -FX) including autonegotiation [27].

1995	The ISO/IEC publishes a backward compatible MPEG-2 Audio (MPEG-2 Part 3) specification – commonly referred to as MP3 – with additional bit and sample rates [36].
1995 Jun.	IETF releases the IPv4 specification "Requirements for IP Version 4 Routers," RFC 1812 [37].
1995 Aug.	IETF releases the first IPsec specification RFC 1825 [38].
1995 Dec.	IETF release the first specification for IPv6 as RFC 1883 [39].
1996 Feb. 14	The Windows 95 Service Pack–1 includes Explorer 2.0 (i.e., built-in TCP/IP networking) [11] [40] [41].
1996 May	HTTP/1.0 is published as RFC 1945 [42].
1997	IEEE 802.3 releases 802.3x full duplex and flow control [27].
1997 Apr.	The Fieldbus Foundation funds the project to develop the "High-Speed Ethernet (HSE) Industrial Ethernet version [43].
1998	IEEE 802.1 publishes the IEEE 802.1D-1998 revision that incorporates IEEE 802.1p with new priority classes [44] and IEEE 802.1Q, which enables VLANs [45].
1998	IEEE 802.3 releases 802.3ac, which extends the maximum frame size to 1522 bytes, to allow 802.1Q VLAN information and 802.1p priority information to be included ("Q-tag") [27].
1998	Founding of the LIN consortium by Audi, BMW, Daimler, Volkswagen, Volvo, Freescale (erstwhile Motorola), and Mentor Graphics (erstwhile Volcano) [46].
1998 Sep. 10	Founding of the MOST corporation by BMW, Daimler, Oasis (now Microchip), and (Harman) Becker [47].
1998 Dec.	IETF publishes the "Internet Protocol, Version 6 (IPv6) Specification," RFC 2460 [48].
1999	IEEE 802.3 releases the 1000BASE-T specification 802.3ab [27].
1999 May	Napster was launched and significantly simplified MP3 music sharing. It was closed in February 2001 [49].
2000 May	Boing delivers its first 747-400, which includes an advanced flight deck display system that uses the Rockwell Collins–developed, Ethernet-based Avionics Systems Network (ASN) as a communication system [50].
2000 Dec. 31	IEC adopts its IEC 61158 standard on fieldbuses. It contains no less than 18 variants. The Ethernet-based variants HSE, EtherNet/IP, and ProfiNet represent three of them [51].
2000	Freescale (formerly Motorola, now NXP), NXP (formerly Philips), BMW, and DaimlerChrysler (today again Daimler) found the FlexRay Consortium [52].
2001 Oct.	DaimlerChrysler (today again Daimler) introduces LIN as the first car manufacturer [53].
2001 Nov.	The first (BMW) car with MOST25 bus and an LVDS-based pixel link goes into production.
2002 Nov.	Release of the IEEE 1588 PTP standard, which had been initiated a few years earlier by Agilent Technologies [54].

2003	IEEE 802.3 releases the first Power over Ethernet (PoE) specification (IEEE802.3af) [27].
2003	The AUTOSAR consortium is founded by BMW, BOSCH, Continental, DaimlerChrysler (today Daimler), Siemens VDO (today Continental), and Volkswagen [55].
2003 Jun. 10	Release of the ARINC Specification 664 Part 2 "Ethernet Physical and Data Link Layer Specification" [56].
2003 Nov.	LIN 1.3 is published [46].
2004	Start of investigations at BMW to use Ethernet as an in-vehicle networking technology.
2004 Feb.	The Metro Ethernet Forum releases the first of a number of standards for the deployment of Carrier Ethernet [57].
2004 Jul.	IEEE 802.3 passes a CFI on "Residential Ethernet" and starts a respective Study Group (SG), i.e., the Audio Video Bridging (AVB) activities [58].
2004 Sep.	IEEE 802.3 releases the first Ethernet in the First Mile (EFM) specification (IEEE 802.3ah) [27].
2005 Apr. 27	First flight of the A380 using an AFDX network for its avionics system (see, e.g., [59] [60]).
2005 Jun. 27	Publication of the ARINC 664 Part 7 specification on "Avionics full-duplex switched Ethernet (AFDX) network" [56].
2005 Nov. 21	The AVB activities are shifted from IEEE 802.3 to IEEE 802.1 [61].
2006	IEEE 802.3 releases the 10GBASE-T specification (IEEE 802.3an) [27].
2006 Feb.	First cars with built-in USB interface for connecting consumer devices are being sold [62] [63].
2006 Aug. 18	IEEE 802.1 releases the 802.1AE specification, also known as MACsec [64]
2006 Nov.	BMW has the first car with a FlexRay bus in production [65].
2007	Toyota introduces the first car with MOST50 [66].
2007 Jul. 20	IEEE 802 confirms the renaming of the 802.3 group from "CSMA/CD (Ethernet)" to "Ethernet" [67].
2008 Jan.	First EMC measurements of BroadR-Reach, today referred to as 100BASE-T1 Ethernet, at BMW.
2008 Oct.	SOP of the BMW 7 series using 100BASE-TX unshielded as a diagnostic interface and using 100BASE-TX shielded for the communication between HU and RSE [68].
2009	The development of FlexRay is seen as completed. The work in the FlexRay Consortium is completed [69] and the specifications are transferred to ISO 17458.
2009 Mar.	The GENIVI Alliance was founded by BMW, Delphi, General Motors, Intel, Magneti Marelli, PSA Peugeot Citroën, Visteon, and Wind River [70].
2009 Aug. 25	The AVnu Alliance is founded by Broadcom, Cisco, Harman, Intel, and Xilinx [71].

2009 Dec. 7	AUTOSAR 4.0 is published and provides means to support Diagnosis-over-IP (DoIP), i.e., Ethernet communication based diagnosis and software flashing via IP and UDP [72].
2010	IEEE 802.3 releases 802.3az on Energy-Efficient Ethernet (EEE) [27].
2010 Jan.	First informal discussion among various car manufacturers and FTZ on UTSP Ethernet [73].
2010 Mar.	BMW internal decision on using BroadR-Reach/100BASE-T1 Ethernet for the next surround view system [73].
2011 Jan.	First discussion between Broadcom, NXP, and BMW on founding the OPEN Alliance [73].
2011 Jan. 31	The IANA assigns the last two available blocks of IPv4 addresses [74]. The number of available IPv4 addresses is thus exhausted.
2011 Mar.	BMW internal decision on using BroadR-Reach/100BASE-T1 Ethernet for the infotainment domain [73].
2011 Aug. 8	The FlexRay Consortium is officially dissolved.
2011 Oct. 15	ISO published the DoIP standard [75].
2011 Nov. 9	NXP, Broadcom, and BMW start the OPEN Alliance. In the same month, C&S, Freescale (now NXP), Harman, Hyundai, Jaguar Landrover, and UNH-IOL join [76].
2011 Nov. 9	NXP announces the development of a BroadR-Reach/100BASE-T1 Ethernet-compliant PHY [77].
2011 Nov. 14	First Ethernet&IP@Automotive Technology Day at BMW in Munich [78].
2011 Sep. 30	The IEEE ratifies and publishes the last of its “Audio Video Bridging” (AVB) standards (IEEE 802.1BA) [79].
2012 Feb.	The Metro Ethernet Forum publishes a suite of specifications as Carrier Ethernet 2.0 [80].
2012 Mar. 15	Call for Interest (CFI) passes for Reduced Twisted Pair Gigabit Ethernet (RTPGE, later called 1000BASE-T1) at IEEE 802.3 [81].
2012 Sep. 19	Second Ethernet&IP@Automotive Technology Day hosted by Continental in Regensburg [82].
2012 Sep.	Audi starts the production of its first car with a MOST150 network [83].
2012 Nov.	IEEE renames the AVB activities as Time-Sensitive Networking (TSN) [84].
2012 Nov. 15	CFI passes for “distinguished minimum latency traffic in a converged traffic environment,” later called Interspersing Express Traffic (IET)/IEEE802.3br, at IEEE 802.3 [85], after it had failed its first attempt on 12 March [86].
2013 Jan.	Start of RTPGE/1000BASE-T1 task force at IEEE 802.3 [87].
2013 Jul.	The LIN standardization is seen as completed. The LIN specifications are transferred to ISO 17987 [88] and the LIN Consortium is dissolved.
2013 Jul. 16	CFI passes for Power over Data Line (PoDL) at IEEE 802.3 [89].
2013 Sep.	SOP of the BMW X5 using BroadR-Reach/100BASE-T1 Ethernet for connecting the cameras to the surround view system [73].

2013 Sep. 25	Third Ethernet&IP@Automotive Technology Day hosted by BOSCH in Stuttgart [90].
2013 Nov.	Acceptance of Interspersing Express Traffic (IET)/IEEE 802.3br Task Force at IEEE 802.3 [91] after failing the attempt in July [92].
2014 Jan.	Start of PoDL Task Force at IEEE 802.3 [93]
2014 Mar. 20	CFI for 1 Twisted Pair 100 Mbps Ethernet (1TPCE) PHY at IEEE 802.3, i.e., the transfer of BroadR-Reach to the IEEE standard 100BASE-T1 [94].
2014 Mar. 20	CFI for Gigabit Ethernet over Plastic Optical Fiber, now called 1000BASE-RH, at IEEE 802.3 [95].
2014 Mar. 31	AUTOSAR Version 4.1 is published and supports TCP, Service Discovery (SD), and the connection to the MAC and PHY layers (including BroadR-Reach/100BASE-T1) [96].
2014 Jun. 9	The OPEN Alliance has more than 200 members [97].
2014 Sep.	Start of 100BASE-T1 Task Force at IEEE 802.3 [98].
2014 Oct. 23	IEEE-SA (fourth) Ethernet&IP@Automotive Technology Day hosted by General Motors in Detroit [99] and organized by IEEE-SA.
2015 Sep.	SOP of 7-series BMW using 100BASE-T1 Ethernet as system bus to connect a variety of ECUs [73].
2015 Jan.	Start of GEPOF/1000BASE-RH Task Force at IEEE 802.3 [100] after failing to move into Task Force in July [101].
2015 May 12	Publication update of the Automotive Ethernet AVB specification [102].
2015 Oct. 27	Fifth Ethernet&IP@Automotive Technology Day hosted by JASPAR in Yokohama [103] and organized by Nikkei BP.
2015 Oct. 14	Among other car manufacturers, Volkswagen and Jaguar Landrover publicly announce the use of BroadR-Reach/100BASE-T1 Ethernet in their cars [104].
2015 Oct. 26	Publication date of 100BASE-T1 specification by IEEE [105]
2016 Jan.	ISO starts Project 21111 Part 1 and 3 on “Road vehicles – In-vehicle Gigabit Ethernet system” with focus on specifications to support the optical Gbps Ethernet standard 1000BASE-RH [106] [107].
2016 Mar. 4	Publication date of the significantly amended IEEE 1722 specification [108].
2016 Mar. 22	OPEN Alliance has more than 300 members [109].
2016 Jun.	The ISO registers ISO 21806 in order to accommodate the completed MOST specifications at ISO.
2016 Jun. 30	Publication date of the 1000BASE-T1 specification by IEEE [110].
2016 Jun. 30	Publication date of the Interspersing Express Traffic (IET) specification by IEEE [111].
2016 Jul. 28	CFI passes at IEEE 802.3 in order to establish a SG to investigate the standardization of a 10 Mbps Ethernet for use in automotive and industrial [112].

2016 Sep. 20 IEEE-SA (sixth) Ethernet&IP@Automotive Technology Day hosted by Renault in Paris [113] and organized by IEEE-SA.

2016 Sep. The ISO project 21111 is renamed from "Road vehicles – In-vehicle Gigabit Ethernet system" to "Road vehicles – In-vehicle Ethernet system" in order to be able to comprise future Automotive Ethernet support specifications for different PHY technologies. The original parts 1 and 3 are split into part 1 to part 4, with the new parts 1 and 2 containing information that is applicable to all Automotive Ethernet PHY variants.

2016 Nov. 10 IEEE 802.3 agrees on requesting to move the 10 Mbps PHY activity for industrial and automotive to Task Force [114]. This effort receives the number IEEE 802.3cg and is expected to be named (a variant of) 10BASE-T1 .

2016 Nov. 10 CFI passes at IEEE 802.3 in order to establish a SG to investigate the standardization of a multi-Gbps Ethernet for use in automotive [115].

References

[1] AT&T, "Milestones in AT&T History" [Online]. Available: www.corp.att.com/history/milestones.html. [Accessed 23 June 2013].

[2] A. Dunn, "The father of invention: Dick Morley looks back on the 40th anniversary of the PLC," 12 September 2008. [Online]. Available: www.automationmag.com/features/the-father-of-invention-dick-morley-looks-back-on-the-40th-anniversary-of-the-plc.html. [Accessed 27 July 2013].

[3] M. Lasar, "The UNIX revolution – thank you, Uncle Sam?," arstechnica, 19 July 2011. [Online]. Available: http://arstechnica.com/tech-policy/2011/07/should-we-thank-for-feds-for-the-success-of-unix/. [Accessed 23 May 2013].

[4] S. Crocker, "Host Software," 7 April 1969. [Online]. Available: http://tools.ietf.org/html/rfc1. [Accessed 4 August 2013].

[5] Wikipedia, "ARPANET," 22 November 2016. [Online]. Available: http://en.wikipedia.org/wiki/ARPANET. [Accessed 23 November 2016].

[6] Wikipedia, "History of the Internet," 16 November 2016. [Online]. Available: http://en.wikipedia.org/wiki/History_of_the_Internet. [Accessed 22 November 2016].

[7] Wikipedia, "Unix," Wikipedia, 16 November 2016. [Online]. Available: http://en.wikipedia/wiki/Unix. [Accessed 22 November 2016].

[8] D. E. Möhr, "Was ist eigentlich EMV? – Eine Definition," 2013. [Online]. Available: www.emtest.de/de/what_is/emv-emc-basics.php. [Accessed 17 December 2013].

[9] R. Metcalfe, R. M. Metcalfe, "The History of Ethernet," 2006. [Online]. Available: www.youtube.com/watch?v=g5MezxMcRmk. [Accessed 26 Febrary 2017.

[10] C. E. Surgeon, *Ethernet: The Definite Guide*, Sebastopol, CA: O´Reilly, 2000, February.

[11] Wikipedia, "Internet Protocol Suite," Wikipedia, 19 November 2016. [Online]. Available: http://en.wikipedia.org/wiki/Internet_protocol_suite. [Accessed 22 November 2016].

[12] S. Djiev, "Industrial Networks for Communication and Control," [Online]. Available: http://anp.tu-sofia.bg/djiev/PDF%20files/Industrial%20Networks.pdf. [Accessed 25 July 2013].

[13] R. M. Metcalfe, D. R. Boggs, C. P. Thacker, and B. W. Lampson, "Multipoint data communication system (with collision detection)," U.S. Patent 4,063,220, 31 March 1975.

[14] D. Boggs and R. Metcalfe, "Ethernet: Distributed Packet Switching for Local Computer Networks," *Communications of the ACM*, vol. 19, no. 7, pp. 395–405, July 1976.

[15] A. L. Russel, "OSI: The Internet That Wasn't," 29 July 2013. [Online]. Available: http://spectrum.ieee.org/computing/networks/osi-the-internet-that-wasnt. [Accessed 21 August 2013].

[16] W. Stallings, "The Origin of OSI," 1998. [Online]. Available: http://williamstallings.com/Extras/OSI.html. [Accessed 12 May 2013].

[17] Wikipedia, "Berkeley Software Distribution," Wikipedia, 11 November 2016. [Online]. Available: http://en.wikipedia.org/wiki/Berkeley_Software_Distribution. [Accessed 22 November 2016].

[18] Avionics Interface Technologies, "ARINC 429 Protocol Tutorial," undated. [Online]. Available: http://aviftech.com/files/2213/6387/8354/ARINC429_Tutorial.pdf. [Accessed 14 July 2013].

[19] U. v. Burg and M. Kenny, "Sponsors, communities, and standards: Ethernet vs. Token Ring in the local area networking business," *Industry and Innovation*, vol. 10, no. 4, pp. 351–375, December 2003.

[20] J. Postel, "User Datagram Protocol," 29 August 1980. [Online]. Available: http://tools.ietf.org/html/rfc768. [Accessed 4 August 2013].

[21] Digital Equipment Corporation, Intel Corporation, Xerox Corporation, "The Ethernet, a Local Area Network. Data Link Layer and Physical Layer Specifications, Version 1.0," 30 September 1980. [Online]. Available: http://ethernethistory.typepad.com/papers/EthernetSpec.pdf. [Accessed 23 June 2013].

[22] Wikipedia, "Ethernet," Wikipedia, 18 November 2016. [Online]. Available: http://en.wikipedia.org/wiki/Ethernet. [Accessed 23 November 2016].

[23] Information Sciences Institute University of Southern California, "Transmission Control Protocol," September 1981. [Online]. Available: http://tools.ietf.org/html/rfc793. [Accessed 4 August 2013].

[24] Information Sciences Institute University of Southern California, "Internet Protocol," September 1981. [Online]. Available: http://tools.ietf.org/html/rfc791. [Accessed 4 August 2013].

[25] J. Postel, "Simple Mail Transport Protocol," August 1982. [Online]. Available: http://tools.ietf.org/html/rfc821. [Accessed 4 August 2013].

[26] Digital Equipment Corporation, Intel Corporation, Xerox Corporation, "The Ethernet, a Local Area Network. Data Link Layer and Physical Layer Specifications, Version 2.0," November 1982. [Online]. Available: http://decnet.ipv7.net/docs/dundas/aa-k759b-tk.pdf. [Accessed 23 June 2013].

[27] Wikipedia, "IEEE 802.3," Wikipedia, 20 October 2016. [Online]. Available: http://en.wikipedia.org/wiki/IEEE_802.3. [Accessed 22 November 2016].

[28] CiA, "CAN history," 2013. [Online]. Available: www.can-cia.de/can-knowledge/can/can-history/. [Accessed 2 August 2016].

[29] Condor Engineering, "AFDX Protocol Tutorial," May 2005. [Online]. Available: www.cems.uwe.ac.uk/~ngunton/afdx_detailed.pdf. [Accessed 15 July 2013].

[30] R. Braden, "RFC 1122," October 1989. [Online]. Available: http://tools.ietf.org/pdf/rfc1122.pdf. [Accessed 23 June 2013].

[31] R. Braden, "RFC 1123," October 1989. [Online]. Available: tools.ietf.org/pdf/rfc1123.pdf. [Accessed 26 June 2013].

[32] P. Barford, "The World Wide Web," University of Wisconsin, 11 September 2008. [Online]. Available: http://pages.cs.wisc.edu/~pb/640/web.ppt. [Accessed 6 August 2013].

[33] Wikipedia, "TIA/EIA-568," Wikipedia, 16 October 2016. [Online]. Available: https://en.wikipedia.org/wiki/TIA/EIA-568. [Accessed 19 November 2016].

[34] Automotive Electronics Council, "AEC History" [Online]. Available: www.aecouncil.com/AECHistory.htm. [Accessed 17 December 2013].

[35] iLocus, "The 10 That Established VoIP," September 2007. [Online]. Available: www.ilocus.com/sites/default/files/The%2010%20That%20Established%20VoIP.pdf. [Accessed 8 August 2013; no longer available, see https://de.scribd.com/document/275142588/The-10-That-Established-VoIP].

[36] ISO/IEC, "ISO/IEC 13818–3:1995, Information technology – Generic coding of moving pictures and associated audio information – Part 3: Audio," ISO, Geneva, 1995.

[37] F. Baker, "Requirements for IP Version 4 Routers," June 1995. [Online]. Available: http://tools.ietf.org/pdf/rfc1812.pdf. [Accessed 23 June 2013].

[38] R. Atkinson, "Security Architecture for the Internet Protocol," August 1995. [Online]. Available: http://tools.ietf.org/html/rfc1825. [Accessed 2 February 2014].

[39] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6)," December 1995. [Online]. Available: http://tools.ietf.org/html/rfc1883. [Accessed 23 November 2013].

[40] Wikipedia, "Windows 95," Wikipedia, 11 November 2016. [Online]. Available: http://en.wikipedia.org/wiki/Windows_95. [Accessed 19 November 2016].

[41] Microsoft, "Verfügbarkeit von Microsoft Windows 95 Service Pack 1-Komponenten," 14 February 2006. [Online]. Available: https://support.microsoft.com/de-de/kb/142794. [Accessed 23 June 2013].

[42] T. Berners-Lee, R. Fielding, and H. Frystyk, "HTTP/V1.0," May 1996. [Online]. Available: http://tools.ietf.org/html/rfc1945. [Accessed 4 August 2013].

[43] Contemporary Controls, "Dick Caro – One of the Most Influential Persons in the Field of Industrial Networking – Part 2," 2006. [Online]. Available: www.ccontrols.com/pdf/Extv6n3.pdf. [Accessed 15 August 2013].

[44] IEEE Computer Society, "IEEE 802.1D-1998: Media Access Control (MAC) Bridges," IEEE, New York, 1998.

[45] IEEE Computer Society, "IEEE 802.1Q-1998: Virtual Local Area Networks (VLANs)," IEEE, New York, 1998.

[46] D. Marsh, "LIN simplifies and standardizes in-vehicle networks," *EDN*, pp. 29–40, 28 April 2005.

[47] A. Grzemba, *MOST, the Automotive Multimedia Network; from MOST 25 to MOST 150*, Poing: Franzis, 2011.

[48] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification," December 1998. [Online]. Available: http://tools.ietf.org/pdf/rfc2460.pdf. [Accessed 23 June 2013].

[49] T. Lamont, "Napster: the Day the Music Was Set Free," 24 February 2013. [Online]. Available: www.theguardian.com/music/2013/feb/24/napster-music-free-file-sharing. [Accessed 30 October 2013].

[50] J. W. Ramsey, "Boeing's 767–400ER: Ethernet Technology Takes Wing," 1 May 2000. [Online]. Available: www.aviationtoday.com/av/commercial/Boeings-767–400ER-Ethernet-Technology-Takes-Wing_12652.html. [Accessed 2 August 2013].

[51] M. Felser and T. Sauter, "Standardization of Industrial Ethernet – the Next Battlefield?," *Proc. IEEE 5th International Workshop on Factory Communication Systems*, pp. 413–421, 22 September 2004.

[52] U. Niggli, "FlexRay, Eine Übersicht über die neue Datenbus-Generation für den Automobil-Bereich," January 2006. [Online]. Available: https://prof.hti.bfh.ch/uploads/media/flexray.pdf. [Accessed 24 September 2013].

[53] D. Müller, D. Sommer, and S. Stegemann, "Local Interconnect Network (LIN)," 13 November 2009. [Online]. Available: http://prof.beuth-hochschule.de/uploads/media/LIN-Bus_MuellerSommerStegemann.pdf. [Accessed 17 September 2013].

[54] Hirschmann, "Precision Clock Synchronization, The Standard IEEE 1588," 2008. [Online]. Available: www.belden.com/docs/upload/precision_clock_synchronization_wp.pdf. [Accessed 28 October 2013].

[55] F. Leitner, "AUTOSAR AUTomotive Open System ARchitecture," 2007. [Online]. Available: www.inf.uni-konstanz.de/soft/teaching/ws07/autose/leitner-autosar.pdf. [Accessed 9 October 2013, no longer available].

[56] AIM GmbH, "AFDX Workshop," March 2010. [Online]. Available: www.afdx.com/pdf/AFDX_Training_October_2010_Full.pdf. [Accessed 13 July 2013].

[57] Metro Ethernet Forum, "MEF Technical Specifications," Metro Ethernet Forum, January 2011. [Online]. Available: www.mef.net/carrier-ethernet/technical-specifications. [Accessed 31 October 2016].

[58] IEEE 802.3, "Residential Ethernet, IEEE 802.3 Call for Interest," July 2004. [Online]. Available: http://grouper.ieee.org/groups/802/3/re_study/public/200407/cfi_0704_1.pdf. [Accessed 12 October 2013].

[59] AIM GmbH, "What is AFDX," 2016. [Online]. Available: www.afdx.com/. [Accessed 31 July 2016].

[60] J. Henley, "At Seven Storeys High, £6bn to Develop and with a Wingspan of 80 Metres – the A380 Makes Its Maiden Flight," *The Guardian*, www.theguardian.com/business/2005/apr/28/theairlineindustry.travelnews1, 27 April 2005.

[61] IEEE 802.3, "IEEE 802.3 Residential Ethernet Study Group Homepage," 10 January 2006 (closed). [Online]. Available: http://grouper.ieee.org/groups/802/3/re_study/. [Accessed 11 October 2013].

[62] just-auto, "USA: Cars to Get USB Ports Soon – CD Players to Become Obsolete?," 31 October 2005. [Online]. Available: www.just-auto.com/news/cars-to-get-usb-ports-soon-cd-players-to-become-obsolete_id84660.aspx. [Accessed 15 September 2013].

[63] bookofjoe, "Microsoft Windows Automotive – World's First Production Car with a Factory-Installed USB Port," 12 March 2006. [Online]. Available: www.bookofjoe.com/2006/03/microsoft_windo.html. [Accessed 15 September 2013].

[64] IEEE Computer Society, "IEEE 802.1AE-2006: Media Access Control (MAC) Security," IEEE, New York, 2006.

[65] F. Völkel, "BMW X5: Erstes Auto mit FlexRay Datenbus," 17 August 2006. [Online]. Available: www.tomshardware.de/bmw-x5-flexray-datenbus,testberichte-1546.html. [Accessed 15 September 2013].

[66] H. Schoepp, "In-Vehicle Networking Today and Tomorrow," 17 January 2012. [Online]. Available: www.automotive-eetimes.com/en/in-vehicle-networking-today-and-tomorrow.html?cmp_id=71&news_id=222902008. [Accessed 28 August 2013].

[67] IEEE 802, "Agenda & Minutes (Unconfirmed) – IEEE 802 LMSC Executive Committee Meeting (updated 24 September 2007)," 20 July 2007. [Online]. Available: www.ieee802.org/minutes/jul2007/Minutes%20-%20Friday%20July%2020%202007.pdf. [Accessed 26 June 2016].

[68] K. Matheus, "OPEN Alliance – Stepping Stone to Standardized Automotive Ethernet," in *2nd Ethernet&IP@Automotive Technology Day*, Regensburg, 2012.

[69] K. R. Avinash, P. Nagaraju, S. Surendra, and S. Shivaprasad, "FlexRay Protocol based an Automotive Application," *International Journal of Emerging Technology and Advanced Engineering*, pp. 50–55, May 2012.

[70] B. Wuelfing, "CeBIT 2009: BMW and Partners Found GENIVI Open Source Platform," 3 March 2009. [Online]. Available: www.linuxpromagazine.com/Online/News/CeBIT-2009-BMW-and-Partners-Found-GENIVI-Open-Source-Platform. [Accessed 10 October 2013].

[71] Business Wire, "AVnu Alliance Launches to Advance Quality of Experience for Networked Audio and Video," 25 August 2009. [Online]. Available: www.businesswire.com/news/google/20090825005929/en. [Accessed 8 October 2013].

[72] AUTOSAR, "Requirements on Ethernet Support in AUTOSAR, Release 4.0, Revision 1," 7 December 2009. [Online]. Available: www.autosar.de/download/R4.0/AUTOSAR_SRS_Ethernet.pdf. [Accessed 9 October 2013, no longer available].

[73] K. Matheus, "Structural Support for Developing Automotive Ethernet," in *3rd Ethernet&IP@Automotive Technology Day*, Stuttgart, 2013.

[74] Number Resource Organization, "Free Pool of IPv4 Address Space Depleted," 3 February 2011. [Online]. Available: www.nro.net/news/ipv4-free-pool-depleted. [Accessed 23 November 2013].

[75] ISO, "ISO 13400–1:2011 – Road vehicles – Diagnostic Communication over Internet Protocol (DoIP)," ISO, Geneva, 2011.

[76] PR Newswire, "Broadcom, NXP, Freescale, and Harman Form OPEN Alliance Special Interest Group," 9 November 2011. [Online]. Available: www.prnewswire.com/news-releases/broadcom-nxp-freescale-and-harman-form-open-alliance-special-interest-group-133514928.html. [Accessed 9 November 2011].

[77] marketwire, "NXP Develops Automotive Ethernet Transceivers for In-Vehicle Networks," 9 November 2011. [Online]. Available: www.marketwired.com/press-release/nxp-develops-automotive-ethernet-transceivers-for-in-vehicle-networks-nasdaq-nxpi-1584249.htm. [Accessed 9 November 2011].

[78] I. Riches, "BMW 1st Ethernet & IP @ Automotive Techday – Momentum Achieved," 14 November 2011. [Online]. Available: www.strategyanalytics.com/strategy-analytics/blogs/automotive/powertrain-body-chassis-safety/powertrain-body-chassis-and-safety/2011/11/16/bmw-1st-ethernet-ip-@-automotive-techday–momentum-achieved. [Accessed 18 November 2011].

[79] Wikipedia, "Audio Video Bridging," Wikipedia, 8 November 2016. [Online]. Available: http://en.wikipedia.org/wiki/Audio_Video_Bridging. [Accessed 23 November 2016].

[80] P. Marwan, "Carrier Ethernet 2.0 ist an den Start gegangen," 28 February 2012. [Online]. Available: www.zdnet.de/41560473/carrier-ethernet-2–0-ist-an-den-start-gegangen/. [Accessed 11 August 2013].

[81] S. Carlson, T. Hogenmüller, K. Matheus, T. Streichert, D. Pannel, and A. Abaye, "Reduced Twisted Pair Gigabit Ethernet Call for Interest," March 2012. [Online]. Available: www.ieee802.org/3/RTPGE/public/mar12/CFI_01_0312.pdf. [Accessed 15 March 2012].

[82] C. Wirth, "Die Netzwerktechnologie Ethernet verspricht eine neue Zukunft im Auto," 10 January 2013. [Online]. Available: www.oth-regensburg.de/fakultaeten/informatik-und-mathematik/nachrichten/einzelansicht/news/die-netzwerktechnologie-ethernet-verspricht-eine-neue-zukunft-im-auto.html. [Accessed 11 October 2013].

[83] MOST Corporation, "MOST150 Inauguration in Audi A3," *MOST Informative*, no. 8, p. 2, October 2012.

[84] IEEE 802.1, "802.1 Plenary -11/2012 San Antonio Closing," November 2012. [Online]. Available: www.ieee802.org/1/files/public/minutes/2012–11-closing-plenary-slides.pdf. [Accessed 30 October 2013].

[85] IEEE 802.3, "Approved Minutes IEEE 802.3 Ethernet Working Group PLENARY," 12–15 November 2012. [Online]. Available: www.ieee802.org/3/minutes/nov12/minutes_1112.pdf. [Accessed 26 June 2016].

[86] IEEE 802.3, "Approved Minutes IEEE 802.3 Ethernet Working Group PLENARY," 12–15 March 2012. [Online]. Available: www.ieee802.org/3/minutes/mar12/minutes_0312.pdf. [Accessed 26 June 2016].

[87] IEEE 802.3, "Approved Minutes IEEE 802.3 Ethernet Working Group PLENARY," 14–18 November 2012. [Online]. Available: www.ieee802.org/3/minutes/nov12/minutes_1112.pdf. [Accessed 7 October 2013].

[88] Consortiuminfo, "Local Interconnect Network Consortium (LIN)," 8 July 2013. [Online]. Available: www.consortiuminfo.org/links/linksdetail2.php?ID=428. [Accessed 2 August 2016].

[89] D. Dwelley, "Power over Data Line, Call for Interest," 16 July 2013. [Online]. Available: www.ieee802.org/3/1PPODL/public/jul13/CFI_01_0713.pdf.

[90] Hanser Automotive, "Rückblick: 3rd Ethernet&IP@Automotive Technology Day," Hanser Automotive Sonderheft Automotive Networks, November 2011.

[91] IEEE 802.3, "Approved Minutes IEEE 802.3 Ethernet Working Group PLENARY," 14 November 2013. [Online]. Available: www.ieee802.org/3/minutes/nov13/1113_minutes.pdf. [Accessed 26 June 2016].

[92] IEEE 802.3, "Approved Minutes IEEE 802.3 Ethernet Working Group PLENARY," 15–18 July 2013. [Online]. Available: www.ieee802.org/3/minutes/jul13/minutes_0713.pdf. [Accessed 26 June 2016].

[93] IEEE 802.3, "IEEE 802.3bu 1-Pair Power over Data Lines (PoDL) Task Force Public Area," 2016 (Continuously updated). [Online]. Available: www.ieee802.org/3/bu/public/index.html.

[94] T. Hogenmüller, S. Abbenseth, S. Buntz, K. M. Albert Kuo, and H. Zinner, "Call for Interest for 1 Twisted Pair 100 [C] Mbit/s Ethernet," March 2014. [Online]. Available: www.ieee802.org/3/cfi/0314_2/CFI_02_0314.pdf. [Accessed 26 June 2016].

[95] C. Pardo, T. Lichtenegger, A. Paris, and H. Hirayama, "Gigabit over Plastic Optical Fibre; Call for Interest," March 2014. [Online]. Available: www.ieee802.org/3/GEPOFSG/public/CFI/GigPOF%20CFI%20v_1_0.pdf. [Accessed 26 June 2016].

[96] AUTOSAR, "Specification of Ethernet Interface, Release 4.1., Revision 3," 31 March 2014. [Online]. Available: www.autosar.org/fileadmin/files/releases/4–1/software-architecture/communication-stack/standard/AUTOSAR_SWS_EthernetInterface.pdf. [Accessed 3 August 2016].

[97] OPEN Alliance, "World's Leading Auto Makers, Tier Ones and Tech Companies Partner to Drive Wide Scale Adoption of Automotive Ethernet," 9 June 2014. [Online]. Available: http://opensig.org. [Accessed 26 June 2016].

[98] IEEE 802.3, "IEEE 802.3bw 100BASE-T1 Task Force Public Area," 17 August 2015 (closed). [Online]. Available: www.ieee802.org/3/bw/public/index.html. [Accessed 26 June 2016].

[99] IEEE-SA, "2014 IEEE-SA Ethernet&IP@Automotive Technology Day," 23 October 2014. [Online]. Available: www.facebook.com/events/1533916296821800/. [Accessed 25 June 2016].

www.cargeek.ir

[100] IEEE 802.3, "IEEE 802.3bv Gigabit Ethernet Over Plastic Optical Fiber Task Force Public Area," 2016 (continuously updated). [Online]. Available: www.ieee802.org/3/bv/public/index.html.

[101] IEEE 802.3, "Approved Minutes IEEE 802.3 Ethernet Working Group PLENARY," 15–18 July 2016. [Online]. Available: www.ieee802.org/3/minutes/jul14/0714_minutes.pdf. [Accessed 26 June 2016].

[102] G. Bechtel, B. Gale, M. Kicherer, and D. Olsen, "Automotive Ethernet AVB Functional and Interoperability Specification Revision 1.4," 12 May 2015. [Online]. Available: http://avnu.org/wp-content/uploads/2014/05/AutoCDSFunctionalSpec-1_4-public_with_legal_notices.pdf. [Accessed 7 July 2016].

[103] B. P. Nikkei, "5th annual IEEE-SA Ethernet&IP@Automotive Technology Day," 27–28 October 2015. [Online]. Available: http://techon.nikkeibp.co.jp/info/ieee/regist/en.html. [Accessed 26 June 2016].

[104] OPEN Alliance, "Automotive Ethernet Hits the Road in Wide Range of New Vehicles," 15 October 2015. [Online]. Available: http://opensig.org/news/. [Accessed 26 June 2016].

[105] IEEE-SA, "IEEE 802.3bw-2015: Amendment 1 Physical Layer Specifications and Management Parameters for 100 Mb/s Operation over a Single Balanced Twisted Pair Cable (100BASE-T1)," IEEE-SA, New York, 2015.

[106] ISO, "ISO/AWI 21111–1 Road Vehicles – In-Vehicle Gigabit Ethernet System – Part 1: General Requirements of Gigabit Ethernet System and Physical and Data-Link Layer Requirements of Optical Gigabit Ethernet System," 4 January 2016. [Online]. Available: www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=69923. [Accessed 26 June 2016].

[107] ISO, "ISO/AWI 21111–3 Road Vehicles – In-Vehicle Gigabit Ethernet System – Part 3: General Requirements and Test Methods of Optical Gigabit Ethernet Components," 25 January 2016. [Online]. Available: www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=70297. [Accessed 26 June 2016].

[108] IEEE, "IEEE 1722–2016: IEEE Standard for a Transport Protocol 2 for Time-Sensitive Applications in 3 Bridged Local Area Networks," IEEE-SA, New York, 2016.

[109] OPEN Alliance, "Non-Profit Alliance Reached a New Milestone: Over 300 Members," 22 March 2016. [Online]. Available: http://opensig.org. [Accessed 26 June 2016].

[110] IEEE-SA, "IEEE 802.3bp-2016: Amendment 4 Physical Layer Specifications and Management Parameters for 1 Gb/s Operation over a Single Twisted Pair Copper Cable," IEEE, New York, 2016.

[111] IEEE-SA, "IEEE 802.3br-2016: Specification and Management Parameters for Interspersing Express Traffic," IEEE, New York, 2016.

[112] L. Winkel, M. McCarthy, D. Brandt, G. ZImmerman, D. Hoglund, K. Matheus, and C. DiMinico, "10 Mb/s Single Twisted Pair Ethernet Call for Interest," 26 July 2016. [Online]. Available: www.ieee802.org/3/cfi/0716_1/CFI_01_0716.pdf. [Accessed 12 November 2016].

[113] IEEE-SA, "2016 IEEE Standards Association (IEEE-SA) Ethernet&IP@Automotive Technology Day," 2016. [Online]. Available: http://standards.ieee.org/events/automotive/index.html. [Accessed 26 June 2016].

[114] G. Zimmerman, "IEEE 802.3 10 Mb/s Single Twisted Pair Ethernet (10SPE) Study Group Closing Report," 10 November 2016. [Online]. Available: www.ieee802.org/3/minutes/nov16/1116_10M_stp_close_report.pdf. [Accessed 12 November 2016].

[115] S. Carlson, H. Zinner, K. Matheus, N. Wienckowski, and T. Hogenmüller, "CFI Multi-Gig Automotive Ethernet PHY," 9 November 2016. [Online]. Available: www.ieee802.org/3/ad_hoc/ngrates/public/16_11/20161108_CFI.pdf. [Accessed 12 November 2016].

1 A Brief History of “Ethernet” (from a Car Manufacturer’s Perspective)

1.1 From the Beginning

In 1969, employees at AT&T/Bell Labs developed the first version of Unix. The original intention was to aid the company’s internal development of software on and for multiple platforms, but over time, Unix evolved to be a very widespread and powerful operating system that facilitated distributed computing. An important reason for the successful proliferation of Unix was that, for antitrust reasons, AT&T was neither allowed to sell Unix nor to keep the intellectual property to itself [1]. In consequence, Unix – in source code – was shared with everybody interested.

It was especially, but not only, embraced by universities, and the community that evolved provided the basis for the computing environment we are used to today and in which Ethernet also has its place. At a time when computing was dominated by large, proprietary, and very expensive mainframe computers few people could use, Unix created a demand for Local Area Networking (LAN) while at the same time providing an affordable, common platform for developing it [2]. As one example, a group at the University of California, Berkeley created a Unix derivative. The Berkeley Software Distribution (BSD) was first released in 1978, and its evolutions became as established as the “BSD-style license” attached to it [3]. Another example is the Transmission Control Protocol (TCP). The first version of this, published in 1974, was implemented for Unix by the University of Stanford by 1979 [4]. Later, in 1989, the then up-to-date TCP/IP code for Unix from AT&T was placed in the public domain and thus significantly helped to distribute the TCP/IP Internet Protocol Suite [5].

The advent of Unix represents an important milestone in the early days of computing. It coincides with the point in time in which a significant number of public as well as proprietary research projects were initiated to investigate methods to interchange data locally and at higher speeds than could be provided for by the telephone system [6]. One of the most momentous projects was the one at Xerox PARC. Xerox needed a solution for data transmission between its first personal computer workstations (called “Xerox Alto”), its laser printers, and the early Internet. Thus, Ethernet was invented (1973), patented (1975) [7], and published (1976) [8].

The general opinion (see, e.g., [9]) is that the foundation of Ethernet’s later success was laid almost as early in time as this, because of the following two choices:

1 **Opening the technology to others:** At the time, it was common for computer companies to try to bind customers to their products by using proprietary technologies or

at least restricting competition with the licensing policy of their patents. Xerox held the patents on Ethernet, but there seems to have been an early understanding that they would profit more from the network effects of a widely deployed Ethernet than from selling the technology itself.[1] Seven years after the invention, on 30 September 1980, Xerox published the "DIX Standard" on Ethernet [10] jointly with the Digital Equipment Corporation (DEC) and Intel. They also offered the technology for adoption to the Institute of Electrical and Electronics Engineers (IEEE) 802 group, very shortly after the group had been founded.[2] With several competing technologies being proposed and followed up, it was by no means evident that Ethernet would prevail. But it did, and one of the reasons attributed to this is that Xerox followed a relaxed licensing policy while not trying to dominate the standardization effort [6]. In the authors' view, this is an attitude as little self-evident then as it is today.

2 **Limiting the technical solution to the task at hand:** Ethernet addressed, and still does address, the communication mechanisms needed on the lower one and a half layers of the ISO/OSI layering model only (see Figure 1.4 in Section 1.2.1), at a time when the ISO/OSI layering model had yet to be completed. It provided a container that gets a packet through a network with multiple participants but is as independent from the application layer as possible [11]. Even today, there is still a tendency to define all layers of a communication system. What allegedly provides the advantages of complete control over the whole communication stack generally makes the system less flexible and less adaptable to future, and hence unknown, requirements. Indeed, Ethernet's adaptability has proven itself to the extent that it is now being introduced in a completely different physical and application environment: in automotive.

In the years that followed, the IEEE became the host for the development of Ethernet. In 1983, IEEE 802.3 published the first of many Ethernet Standards, 10BASE-5 for 10 Mbps over thick coax cable [12]. In the same year, already at least 21 companies were mentioned in the trade press to be developing and/or manufacturing Ethernet products [6]. When, on 1 January 1984, the AT&T monopoly ended, the existing installed telephone wiring became usable for competing services and applications [13] and a whole new range of possibilities opened to the networking world. Thus, in 1987, SynOptics, a Xerox spinout, was the first company to prove the feasibility of transmitting Ethernet at 10 Mbps over telephone wires with a proprietary Ethernet product [6]. The IEEE ratified the respective 10BASE-T standard in September 1990. Because of the many other proprietary versions of Ethernet that had evolved in the meantime, standardization of 10BASE-T was not obvious and required some effort. Nevertheless, when successful, it sealed the victory over other networking technologies in the market [14]. Shortly after, in 1993, an optical Ethernet version was developed and published as 10BASE-F.

Meanwhile, the world around Ethernet did not stand still but continued to provide means and create demands for networking. Various evolutions of TCP and IP were developed, and in October 1989, the IETF published the complete set of protocols in the TCP/IP Internet protocol suite [15] [16]. As mentioned, the success of TCP/IP was fueled by AT&T's public domain implementation of TCP/IP on Unix [5]. In 1991, the TIA published a standard for inexpensive UTP wiring: TIA/EIA-568. Even today, it is

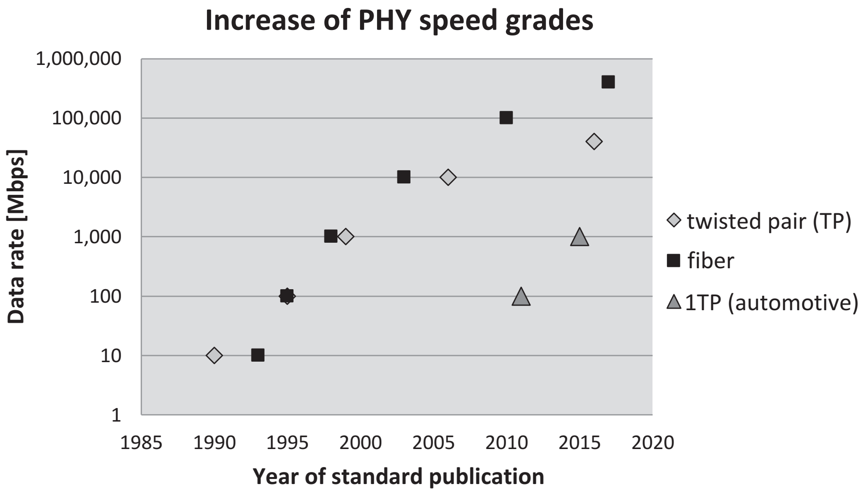

Figure 1.1 Timeline of major PHY speed grades.

impossible to imagine an Ethernet network without the 8P8C/RJ-45 connector described in that standard. The World Wide Web was launched in 1994 [14], and the IETF released a specification for IPv4 routers in June 1995 [17]; the Windows 95 Service Pack–1, released on 14 February 1996, automatically included Microsoft Internet Explorer 2.0 (i.e., built-in TCP/IP networking), bringing the Internet to the masses [18]. Internet Explorer had been available before but needed to be purchased separately.

Subsequently, the IEEE amended and enhanced Ethernet, proving Ethernet's adaptability. First, IEEE 802.3 added, and continues to add, new speed grades. Figure 1.1 gives an overview of the increase of data rates for copper and fiber optical channels. The largest data rates envisioned today are 40 Gbps for transmission for twisted pair cables and 400 Gbps for optical communication. Figure 1.2 gives an overview of all Ethernet Physical Layer (PHY) variants developed or under development. It is noticeable that many of the new developments no longer simply increase the previous data rate by a factor of 10 but that the market is diversified with many in-between speed grades.

Other major developments in IEEE 802.3 are as follows. In 1997, IEEE 802.3 enabled full-duplex communication and flow control to replace the shared media approach prevailing until then. New functionalities that have been added are autonegotiation in 1995, Power over Ethernet (PoE) in 2003, and Energy-Efficient Ethernet (EEE) in 2010. New use cases that have been considered by the IEEE include Ethernet in the First Mile (EFM, 2004; see also Section 1.2.4), Ethernet over copper backplane (2007), and finally, in 2013, a good 15 years after the respective activity had been started for data centers, IEEE 802.3 set up a task force to develop a Reduced Twisted Pair Gigabit Ethernet (RTPGE) suitable for automotive.

In addition to the directly PHY-related activities, the IEEE has worked on, and is still working on, Quality of Service (QoS) schemes for Ethernet and other management functions. In Ethernet, basically the only quality control provided is a CRC check

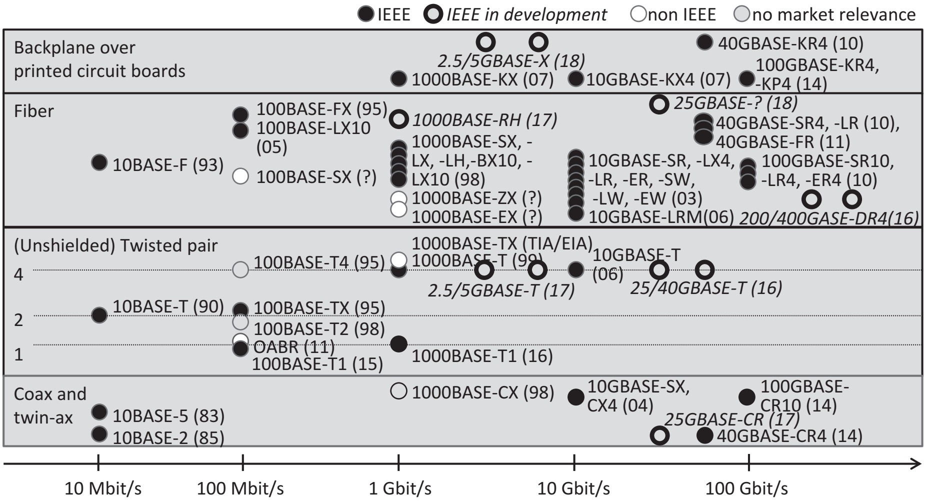

Figure 1.2 Year 2016 overview of Ethernet PHY variants. The (expected) year of release is in brackets, when known [19].

at the receiver, which has no other consequences than offering the possibility to discard the packets with detected errors. A pure IEEE 802.3 measure was taken in 1998, when IEEE 802.3 agreed on a packet extension to incorporate an IEEE 802.1Q header consisting of 802.1 Virtual LAN (VLAN) and priority information. Another important concept was established in 2011, when the IEEE (mainly in 802.1) finalized the first set of standards summarized under Audio Video Bridging (AVB). AVB aims at improving the quality of audio and video transmissions over an Ethernet network (for more details, see Section 5.1). At the time of writing in 2016, further enhancements on the AVB/QoS functionalities were still being standardized under the name of Time-Sensitive Networking (TSN).

1.2 The Meaning of "Ethernet"

The term "Ethernet" was first used in 1973, the name referring to the belief of nineteenth-century physicists that there is a passive medium between Sun and Earth that allows electromagnetic waves to propagate everywhere, which they called the "luminiferous Ethernet." The coax used for the inventors' communication system was equally passive, and they also intended their data packets to go everywhere [14].

Nevertheless, at first, the IEEE did not officially adopt the name (although, unofficially, it did). As an open standards body, the IEEE did not want to give the impression of favoring any company in particular. Despite the fact that Xerox had relinquished its trademark on the name, IEEE 802.3 was called "Carrier Sense Multiple Access with Collision Detection" (CSMA/CD) instead [11]. The official renaming of the IEEE 802.3 efforts into "Ethernet" did not happen until 2007 [20].

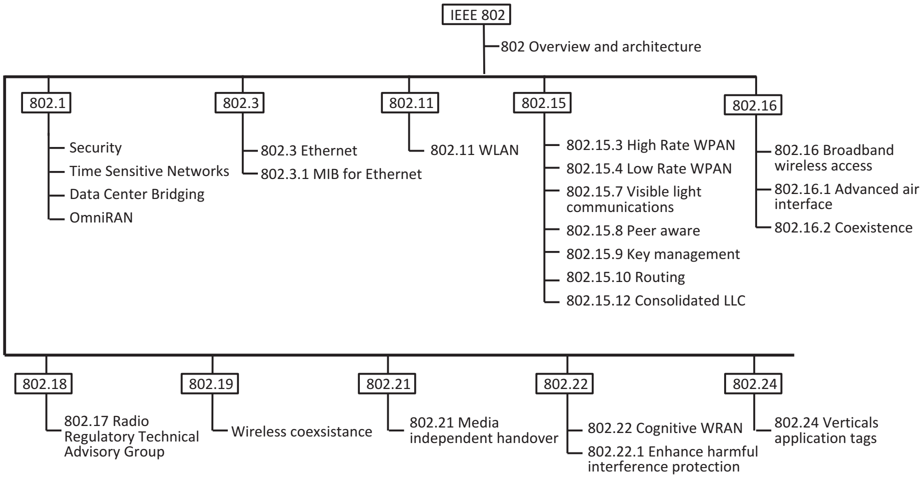

Figure 1.3 The ongoing IEEE 802 standardization activities in 2016 [22].

As a consequence, in the various application fields and industries, the name "Ethernet" is used with different meanings, some of which have very little in common with what is specified in IEEE 802.3. The following sections will therefore address how Ethernet is used in the IEEE, in some other industries, and in the "Automotive Ethernet" discussed in this book.

1.2.1 Ethernet in IEEE

Ethernet is standardized in IEEE 802.3 (see Figure 1.3). This comprises the complete Physical Layer (PHY) and those parts of the Data Link Layer (DLL) that are technology specific, like the packet format and the medium-access method chosen (see Figure 1.4). Various other aspects also in the IEEE standards, e.g., in IEEE 802.1, affect the implementation of an Ethernet-based communication system. While being relevant, these standards are applicable to all technologies addressed in 802 and therefore are not "IEEE Ethernet" specific. This is the same for the Logical Link Control (LLC), whose standardization has been concluded in IEEE 802.2 and whose task is to harmonize various methods of medium access toward the network layer [11] [21].

One of the main inventions of the original Ethernet was sharing the media with the help of a CSMA/CD mechanism. CSMA/CD was based on the ALOHA method, which had been developed at the University of Hawaii a few years earlier as a multiuser access method and which more or less simply proposed retransmissions in case collisions were detected [14]. In the case of CSMA/CD, this was enhanced by additionally establishing whether the channel is occupied before the start of a transmission. If the channel is sensed to be available, the transmitter is allowed to send its packet. Nevertheless, even in this case, collisions can occur, such as when another unit had also sensed the channel as available and started transmitting simultaneously. Both transmitters would detect this

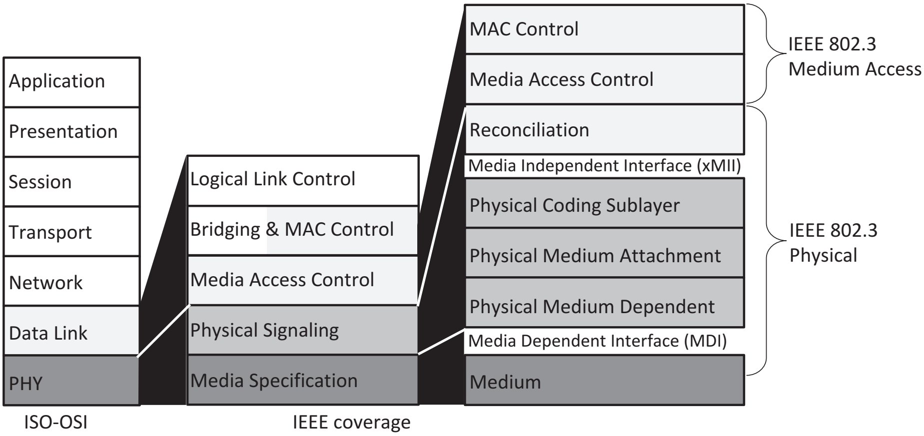

Figure 1.4 Ethernet in IEEE (e.g., [21]).

and, in consequence, go into a random back-off period that would increase its potential length with the number of collisions having occurred for one packet [11].

Today, it is hard to find any Ethernet installation that still uses the CSMA/CD method.[3] The vast majority of Ethernet networks are installed as switched networks with a type of Point-to-Point (P2P) link. In these networks, only the PHYs of two units are connected directly, and switches in the receiving unit forward the packets according to their addressing between the other unit’s internal PHYs.[4] The so-called full-duplex[5] operation provides significant advantages in terms of timing and supported link segment lengths [11], so that today, the CSMA/CD mode has become obsolete. Also, in “full duplex,” the MAC is responsible for receiving and transmitting packets. With full duplex, a new sublayer was added: the MAC Control (2× Control!). The general purpose of the MAC Control layer was to allow for the interception of Ethernet packets in the case of specific requirements. In the case of full duplex, it enables flow control. In order to allow for limited resources in terms of the buffering and switching bandwidth, the MAC Control provides the mechanisms to decide when packets are being sent [21].

The most pronounced and stable element of Ethernet is the Ethernet frame/Ethernet packet (see Figure 1.5). The packet starts with a preamble and the Start Frame Delimiter (SFD), which together help synchronize incoming data in the case of CSMA/CD operation. With CSMA/CD no longer deployed, they have become obsolete but are kept for backward compatibility reasons. Starting with 100BASE-TX, more complex signal encoding has been used, which allows for the deployment of special symbols to detect the beginning and end of a packet.

	Preamble	SFD	Destination MAC address	Source MAC address	Optional 802.1Q Header	Length or Ethertype	Payload/LLC	CRC/ FCS	Inter Frame Gap
Bytes:	7	1	6	6	4	2	42/46-1500	4	min.12

Figure 1.5 Elements of an Ethernet frame/packet.

Each Ethernet interface[6] is assigned a unique serial number consisting of 48 bits, often referred to as the "MAC address" or the "hardware address." Following the preamble, every packet contains information on where the packet is to be sent and the device that sent it, using the respective MAC addresses. End node MACs initially only read up to the destination address to evaluate whether a packet is intended for this end node (as direct-, multi-, or broadcast). If the address matches, the packet is read completely; if the address does not match, the packet is ignored. Switch nodes evaluate both the destination address, deciding which port to send the packet to, and the source address, remembering for future incoming packets on which port to find the addressee with that address. This means that there is normally a learning period after start-up in a switched Ethernet network.

The next four bytes represent an optional IEEE 802.1Q header. The first two bytes identify that this indeed is an 802.1Q header. The remaining two provide the Tag Control Information (TCI) and are divided into three bits for the priority information according to the 802.1p standard, one bit representing the Drop Eligible Identifier (DEI) and 12 bits for the Virtual LAN identifier, which specifies to which Virtual LAN (VLAN) the packet belongs [23]. VLANs represent an important concept for partitioning a physical LAN into various logical domains on layer 2 (see Section 5.2).

The next field indicates either the length of the packet or the Ethertype. The Ethertype states what type of data to expect in the payload in respect to the higher layers. It covers content like IP (v4 or v6) or certain AVB packets but also various proprietary types that have accumulated over time. Ethernet had been designed as a container for whatever data needs to be transmitted; for example, several of the Industrial Ethernet variants – e.g., Profinet, EtherCat, Sercos, Powerlink, High-Speed Ethernet (HSE) – have their own Ethertypes (see Section 1.2.2). The IEEE 802.1Q identifier mentioned above has the Ethertype 0x8100. A list of Ethertypes is maintained by the IEEE [24]. When the field represents the length, the content is a number equal to or less than 1500 (see next paragraph). In this case, the IEEE 802.3 LLC protocol can be used to identify the type of data being transmitted.

The payload has a minimum size of 42 bytes when the 802.1Q header is present and 46 bytes when it is not.[7] Should the data needing to be sent be shorter than the minimum, then the remaining bytes are filled with padding. The maximum payload length is 1500 bytes. Note that the payload represents user data only from a layer 2 perspective. Various headers from other layers, like the IP or User Datagram Protocol (UDP) headers, will further reduce the bytes available for the actual application.

Finally, the packet is terminated with a Cyclic Redundancy Check (CRC) called the Frame Check Sequence (FCS)). The FCS is 32 bit long and checks the integrity of the various bits of the packet (other than preamble and SFD). Following the packet there must be an interframe gap of a minimum of 12 bytes. With a fully loaded payload this means that the header/payload efficiency is larger than 97%.

Table 1.1 provides an overview of the main components of Ethernet and how they have changed over time. As has been visualized in Figure 1.2, Ethernet has been developed for various media and almost all but the original one are being addressed today. As a consequence of higher data rates and advancements in signal processing, the physical

Table 1.1 Comparison of the four main Ethernet components as defined in 10BASE-5 and the “IEEE Ethernet” today

	Ethernet in 10BASE-5	IEEE Ethernet today
Packet	26 byte overhead, 46–1500 byte payload	Optional 4 bytes for 802.1Q header added
MAC	CSMA/CD, best-effort traffic without acknowledgments	Full duplex and flow control, best-effort traffic without acknowledgments
Signaling	Manchester encoding	Various, e.g., PAM-2/3/4/5, DSQ128, NRZ
Media	Coax	TP, fiber, backplane, Twinax

signaling has changed with the media and has also been standardized in various forms. The original media access mechanism vanished. Nevertheless, the principle that Ethernet performs no quality control in form of acknowledgments or retransmits, as well as its “container”-function, has been kept. If needed, retransmits have to be initiated on higher layers. Likewise the Ethernet packet has remained almost unchanged, with only the addition of the optional 802.1Q header.

1.2.2 Ethernet in Industrial Automation

Communication in industrial automation is generally structured hierarchically (see also Figure 1.6). The lowest level of communication happens between sensor or actuator and the low-level controller [25] [26]. The amount of data transmitted with every cycle can consist of a few bits only. Nevertheless, the communication needs to be cost efficient and the response time short. Cycle times for tasks like motion control can be much less than 1 ms with a synchronization accuracy within 1 μs [27]. At a machine level, more

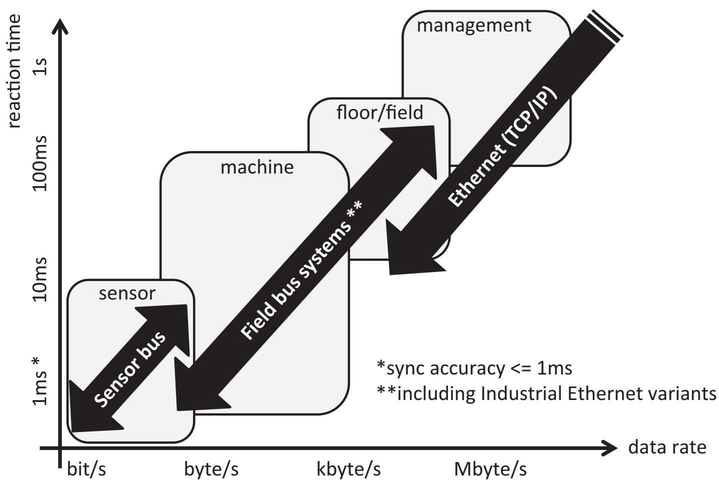

Figure 1.6 Hierarchical approach to factory communication [35] [27].

intelligent field devices like I/O stations, operator panels, and Programmable Logic Controllers (PLCs) exchange data. For most tooling machines or remote Input/Output (I/O) a response time of below 10 ms is required. At the floor (or “field”) level, automation and operator stations communicate with PCs. A response time of 100 ms is sufficient for activities like process monitoring and thus most processes in process automation and building control [27]. Often the floor level is subdivided into smaller “cells” and larger “areas.” This allows the separation of critical from not-so-critical cells and, in the case of issues, enables them to be isolated as well as repaired without affecting the whole production. At the highest, management-level orders, reports, quality statistics, etc. are handled. The requirements for the reaction time are less critical, while the packet size and amount of data increase.

The process of industrialization is the foundation of wealth in occidental society. Hence, right from the beginning of the industrial revolution, efforts have been made to improve and optimize production processes. Naturally, the possibilities of computerization were explored from the early days and the foundations for hierarchical communication were laid in the early 1970s. After the 1960s had brought a number of inventions impacting industrial manufacturing – mini computers, robots, computer/Numerically Controlled (NC) machines, and especially Programmable Logic Controllers (PLCs) – there was a need for efficient communication between the units as well as the possibility for decentralizing their control [28]. It was found that decentralization improved the quality and availability of process observation and control as well as unburdened the central computer. At the same time it removed the need to use a star topology, and thus reduced the amount of cabling [29]. The first commercially available distributed computer control systems were introduced by Honeywell and Yokogawa in 1975 [28].

The rest of industry followed and in the 1980s every company in the automation business seemed to have developed their own “fieldbus”[8] system in order to support the respective communication in manufacturing plants. The large number of fieldbus variants (>50 [30]) nevertheless did not appeal to the customers. In the case of technical problems manufacturing plant owners need access to replacements fast – potentially from a different vendor – to minimize the risk and impact of downtimes. In consequence, suppliers published their specifications [31], which helped to establish fieldbus systems in industrial automation. Up till today fieldbus connected nodes represent the majority of new as well as existing nodes in industrial plants [32]. At the same time, efforts toward standardization were made. The outcome of those efforts is, however, a somewhat double-edged sword: When the International Electrotechnical Commission (IEC) finally adopted its IEC 61158 standard on 31 December 2000, it contained no less than 18 variants [27]. The possibility to have interoperable solutions in general and the possibility to have perfectly fitting solutions for different use cases, was obviously more important and more advantageous than to have a single solution that covers all [33]. The respective standardization efforts in IEEE (802.4) were finally disbanded in 2004 [34].[9]

Fieldbusses can fulfill very small reaction time requirements (see also Figure 1.6). Investigations into the use of fieldbus technologies showed that it is advantageous to use one technology only [36]. Nevertheless, many publications mention the additional

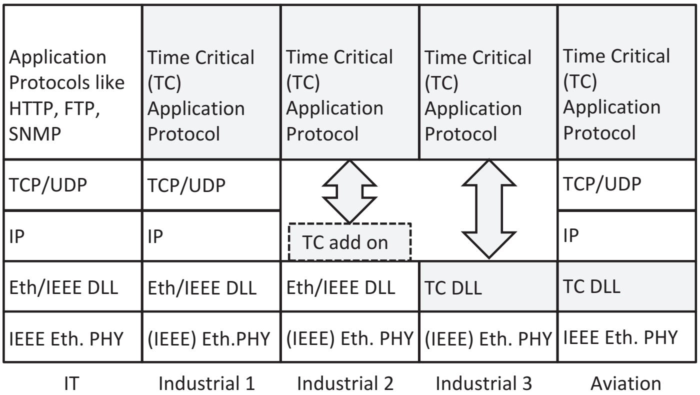

Figure 1.7 Conceptual real-time variants in Industrial Ethernet [26] [27] [37].

use of separate sensor busses for cost reasons (e.g., [35]). On top, the standard Ethernet TCP/IP is used to integrate the management level, which makes it three technologies at minimum. The desire for seamless communication over all hierarchy levels and parts of the production process for complexity and cost reasons is easy to understand, and this made "Ethernet," being part of the system anyway, an obvious choice. Standard Ethernet TCP/IP is nevertheless nondeterministic and reaction times can be above 100 ms, although there are simple means to reduce this, like using UDP instead of TCP or restricting the possible traffic in local sections of the network. With the resulting reaction time of 10 ms [26] a significant number of applications in industrial automation can be covered.

To make Ethernet (even) more suitable for real-time applications and to fulfill various additional requirements on robustness, functional safety, high availability, and security combined with low latency, "Industrial Ethernet"[10] solutions were developed. Figure 1.7 shows the different concepts behind them. In the simplest case a protocol specifically catering for time-critical use cases is used on the application layers ("Industrial 1"). The next option ("Industrial 2") is to have the time-critical traffic bypass the IP and TCP/UDP layers and to directly communicate with the data link layer. The reaction time can thus potentially be shortened down to 1 ms [26]. This bypass concept is also used in IEEE 802.1 Audio Video Bridging (AVB) and is described in more detail in Section 5.1.

In the last variant depicted ("Industrial 3") the data link layer is redefined in order to accommodate the real-time requirements directly in the MAC. This implies the most significant changes that might even affect the implementation in hardware down to the PHY. Even though the aviation industry does not reuse any of the Industrial Ethernet variants for the communication between avionics systems in an aircraft (see also Section 1.2.3), the "Aviation" structure depicted in Figure 1.7 is in the end just another version of "Industrial 3." One of the basic principles behind almost all Industrial Ethernet versions is that the "IT" part of the communication is used for best-effort traffic and that Standard Ethernet hardware is used for the PHY. Note that special variants of cabling

Table 1.2 Market share and volume of industrial networking nodes between 2011 and 2015 [39]

Technology	Number of installed nodes in industrial networks in 2011 (millions)	Projected number of installed nodes, 2015 (millions)	Average number of new nodes per year (millions)	Expected growth, CAGR (%)
Ethernet TCP/IP	3.46	5.40	0.49	11.4
Industrial Ethernet	3.81	6.42	0.65	14.0
Overall Ethernet	7.26	11.82	1.14	12.9
Overall fieldbus	24.04	33.28	2.31	8.5
Overall	31.30	45.10	3.45	9.6

and connectors are generally always used with Industrial Ethernet, for robustness in the physically harsh environment of industrial manufacturing [27].

Note that even with Industrial Ethernet or Aviation adding nonstandard parts to the Data Link Layer (DLL) such an approach is not under discussion for automotive.

Industry thus does not only use a large number of fieldbus variants but also various incompatible types of "Industrial Ethernet." Twenty-nine versions of real-time Ethernet are listed in [38], and seven are mentioned with respect to their market share in [39]. From an automotive perspective, this is surprising, because many incompatible networking technologies result in additional costs and overhead. Even if the costs for the networking technologies are of lower priority in industrial automation than in automotive, it is of high priority for industrial automation customers to be able to obtain replacement units from different vendors to carry out any necessary maintenance and repair work. If all vendors use the same networking technologies, this should also be easier to achieve. Potentially, the industrial automation customers are too diverse and it is only possible in smaller groups, like the Automation Initiative of German Automobile Manufacturers (AIDA), to request uniform solutions.

Table 1.2 shows example market data for industrial automation networking technologies. The numbers indicate that the market shifts from fieldbus solutions toward Ethernet. Nevertheless, the sum ports in the many special Industrial Ethernet solutions is expected to grow more than "standard Ethernet" with TCP/IP. This is an indication that the standard solution is still not seen as the most suitable for the use case.

There are efforts ongoing to change this. In order to be able to use IEEE standardized Ethernet better in Industrial Automation, various activities are being initiated and supported in IEEE. For one requirements from Industrial Automation are discussed with the new specifications of Time-Sensitive Networking (TSN) standards in IEEE 802.1 (see also Section 5.1.4). Then IEEE 802.3 concluded in 2016 its specification on Interspersing Express Traffic (IET)\IEEE802.3br [40], a provision to further reduce latency in an Ethernet network by being able to interrupt the transmission of lower priority packets for high priority ones. Last but not least, in July 2016 a CFI was passed at IEEE 802.3 for the start of a study group to investigate the development of a "10Mb/s Single Twisted Pair Ethernet Call for Interest," with the goal to develop a PHY technology that can replace many of the fieldbus variants with a suitable, higher performing Ethernet PHY [41].

1.2.3 Ethernet in Aviation

In a passenger plane four areas of communication can be distinguished by their different networking requirements: (1) communication between the avionics systems; (2) operational communication for avionics system and cabin communication (maintenance, configuration, . . .); (3) air–ground communication; and (4) passenger communication as part of the in-flight entertainment (IFE). The IFE (4) and maintenance and configuration (2) can potentially use a variety of existing networking technologies from other industries, including Ethernet, and will not be discussed further. The air–ground communication (3) is not relevant in the context of this book. Even though (3) is special in the sense that it requires a frequency band to use and worldwide harmonization, standardization has long been realized. As early as 1929, when commercial air traffic emerged, the Aeronautical Radio, Inc. (ARINC) was established for this purpose and started with coordinating the air–ground communication.

The communication area with very specific requirements and of interest for "Aviation Ethernet" is the communication between avionics systems (1). Today, the ARINC also hosts the development and publication of various standards relevant in this area [42]. One of the most commonly used ARINC specifications for the communication between avionics systems is the ARINC 429, which was first released in April 1978 [42] [43]. ARINC 429 allows for the simplex, i.e., unidirectional communication at either 12.5 kbps or 100 kbps over STP cabling with a word size of 32 bits overall, 19 of which represent the data area. One transmitter can be connected to up to 20 receivers. Nevertheless, all receivers wanting to respond to the transmitter require a separate ARINC 429 link. Star, bus, or mixed topologies are in principle all possible, but as transmitter and receiver need to be directly connected (per transmission direction) any slightly more complex communication need quickly leads to a significant amount of wiring. As commercial planes heavily rely on their avionics systems – the first all-electronic fly-by-wire system was introduced into commercial airplane service in 1988 [44] – the cost and weight of the wiring is everything but negligible.

There were thus several reasons to introduce Ethernet-based communication systems in aviation: allow for larger content per packet, allow for larger data rates, and allow for a modular architecture with standardized components (Integrated Modular Avionics [45]) and flexibility in the network in order to optimize aircraft design. It was of interest to be able to share resources, support the increased interdependencies of avionics systems, reduce hardware costs through using less wiring and, by basing the system on Commercial-Off-The-Shelf (COTS) technologies, reduce cabling weight and allow for the integration/reuse of existing communication technologies outside the plane (especially IP and UDP), while at the same time addressing real-time requirements (see, e.g., [46]).

Boeing was first to introduce an avionics system with Ethernet-based communication adapted to the rigors of the environment [47]. The first Boeing 747–400 with the respective advanced flight deck display system was delivered in May 2000 [41]. Boeing also introduced Ethernet in the Boeing 777 for noncritical systems at less than 10 Mbps. Airbus developed the Avionics Full-Duplex Switched Ethernet (AFDX), which saw its maiden flight with the commercial aircraft A380 in April 2005. AFDX was published in two ARINC standards: ARINC 664 part 2 "Ethernet Physical and Data Link Layer Specification" and ARINC 664 part 7 "Avionics Full-Duplex Switched Ethernet (AFDX) network." AFDX (see also Figure 1.7) relies on standard IEEE 802.3 Ethernet PHYs, but uses them with cabling suitable for the specific environment (which in the case of planes means shielded cables with special temperature resilience). The Data Link Layer (DLL) is changed and adapted, in order to achieve the necessary timing and reliability requirements.

In AFDX, the logic communication is based on so-called Virtual Links (VLs) that restrict the communication depending on an identifier. VLs do not define the LAN inside which the traffic can go unrestricted, as is the case for Virtual LANs (VLANs). They define, as the name indicates, the communication partners for every communication in the system. The MAC addressing fields in the AFDX packets are used in a specific way to accommodate this. AFDX thus (also) allows emulation of the communication defined for ARINC 429, while at the same time profiting from the flexibility and reduced cabling of a switched Ethernet network. An additional control is being performed at every receiver by evaluating the VL identifier and the assigned bandwidth of incoming packets against the preconfigured values [48]. For the bandwidth allocation a Bandwidth Allocation Gap (BAG) is defined per VL. The BAG restricts the amount of traffic that can be transmitted in a specific interval and thus has a sort of traffic shaping function.

Another property of AFDX is the support of redundancy. Redundancy is achieved by the use of sequence numbers and the installation of two parallel links for every physical connection and every VL. At the receiver simply just the first packet to arrive with a specific sequence number is processed. Should the redundant packet also arrive at a later time, it is ignored [48]. Furthermore, careful system design assesses upfront whether the network with all latencies, jitters, and buffer sizes will allow the expected communication to pass as expected.

The described efforts show that also the aviation industry is interested in the (cost) advantages that can be realized with the standardization of nondifferentiating functions like networking and the reuse of solutions successful in other industries. As an industry, it nevertheless has extremely harsh safety and security requirements: avionics systems require a "Design Assurance Level" A [49] with less than one error in 10 billion hours airborne time [50] and the respective certification from the authorities. This sometimes impedes the reuse of technologies[11] and is the main reason why the overlap with, e.g., Industrial Ethernet is very small. At the same time, of the industries discussed in the context of Ethernet in this chapter, aviation is the one with the smallest number of manufacturers (see also Table 1.3). This can make standardization easier, but as every manufacturer is used to being a very powerful customer, it can make them less likely to compromise. In consequence, all the aircraft manufacturers listed in July 2016 as using Ethernet-based ARINC 664 part 7 solutions (Airbus, ATR, Boeing,

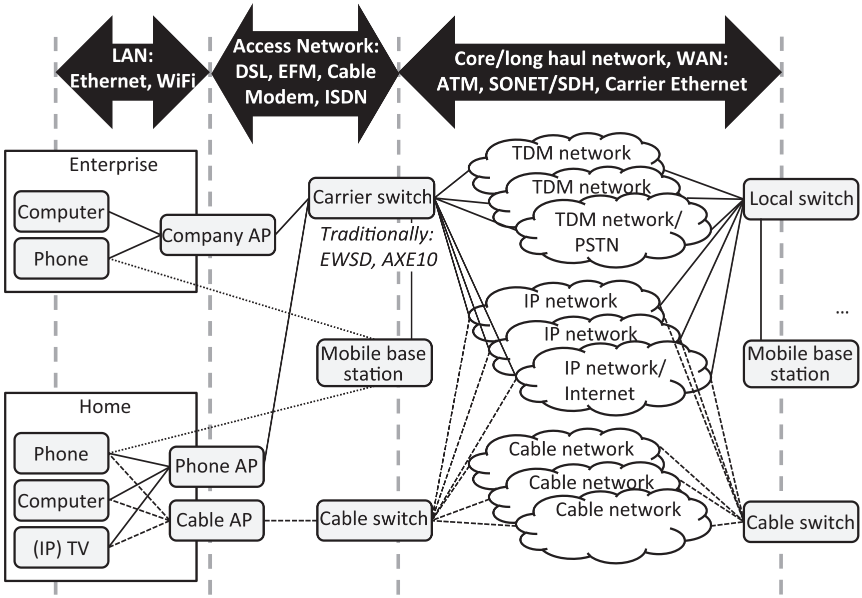

Figure 1.8 Simplified diagram of the relation between the different elements in telecommunications.

Bombardier, Comac, Irkut, Sukhoi, plus the helicopters vendor AgustaWestland [51]) can be expected to use a specific adaptation of the standard.

1.2.4 Ethernet in Telecommunications

For a very long time, enabling voice communication between two parties at different physical locations was the main service provided by telecommunications companies. From the invention of telephony in the middle of the nineteenth century [52] to acknowledging the need for changes, the twenty-first century had to arrive [53]. By now, 2016, these changes have become more than just adaptations. Telecommunication providers are setting target dates for completely abandoning the original, voice-oriented principles of communication – i.e., circuit-switched/Time Division Multiplex (TDM) communication – for packet-switched IP traffic (see, e.g., [54] [55]). To explain the developments that led to this and the relation to Ethernet, a (simplified) distinction is made between the following communication areas: the user domain, the access technologies, and the core telecommunication networks (see also Figure 1.8).

The telecommunication providers laid many of the foundations for today's Information Age, which they now seem to struggle to keep up with [56]. Among other things, they were involved in the development of computers, in how to use them (see Section 1.1), and in physically connecting companies and households to the (telecommunication) network. Nevertheless, the focus of their activities was on improving voice communication. They invested in better voice quality and better coverage. They enabled more simultaneous calls and more connections between the continents and became more cost efficient with improved automation in call switching. The carrier switches that manage the calls between two different subscribers represented a key element in the network and thus were digitized first. The first digital carrier switch was introduced by AT&T in 1965 [13]. Ericsson and Siemens developed their commercial and very successful digital switching products AXE and "Elektronisches Wählsystem Digital" (EWSD) in the late 1970s [57] [58]. However, on the subscriber side the communication stayed analog.

The Information Age was thus driven not by the digitization of the Public Switched Telephone Network (PSTN) but by the military and the computer industry and their desire to share data, which eventually led to the establishment of the Internet. Important milestones in this process were: the development and publication of IPv4 and TCP as RFCs in 1981 [59] [60], the already mentioned break-up of the AT&T monopoly on 1 January 1984 [13], the creation of the Internet Engineering Task Force (IETF) in 1986, the invention of the World Wide Web 1989/90 [61], the decommissioning of the military run ARPANET, and the deregulation of the Internet for commercial use in 1990.

The Internet changed everything. Not only the way people work, live, and communicate but also the paradigm that telephony/voice communication needs to be circuit switched. The application that proved this was Voice over IP (VoIP). There had been very early experiments that showed that voice could be transmitted with data packets. However, the real start of VoIP was in 1995, when the first commercial product allowed real-time, full-duplex voice communication [62]. VoIP first took root in international calls, where the cost savings for users were most significant. The technology improved with the rapidly increasing number of calls and their management in the network became more powerful. With the proliferation of broadband Internet access into the household, VoIP became successful. In 2009, 25% of all voice minutes were using VoIP [63]. By 2013, depending on the actual provider(s), it was possible that a VoIP call was converted to circuit switched for some part of the way or that a regular circuit-switched call was changed into VoIP.

VoIP is one example of the recent changes in telecommunications. In general, services for data/Internet, storage, VPN, voice/VoIP, and video applications are converging with the networks of telephone, Internet, mobile phone, and cable TV providers [64]. The telecommunication providers shift to packet-based communication for competitive reasons: reduction of costs and equipment, possibilities for new services, and complete and seamless integration [54] [65]. The underlying paradigm is the provision of IP-based services, and a logical and cost effective way to realize IP is to use Ethernet as well [66] [67]. As a result the end user can consume hundredfold bandwidth for virtually no added costs.

To deploy Ethernet at the user level (see also Figure 1.8) is simple. Ethernet was primarily designed for company LANs and from there it spread into homes. Enterprise LANs and consumer devices (like PCs, printers, etc.) thus represent the original Ethernet market. The developments specific to telecommunications are "Ethernet in the First Mile" (EFM) for the access network and "Carrier Ethernet"[12] for the core network.

Ethernet in the First Mile (EFM) is an IEEE 802.3 effort, which resulted in the IEEE 802.3ah standard released in September 2004. EFM focuses on two aspects relevant for access networks: longer link distances and additional diagnostic monitoring capabilities. The latter is necessary as – and this is different in the case of LANs – the service provider is likely to be located at a significant physical distance from the subscriber’s potential problem with the network [21]. The “Far-end Operation, Administration, Management” (OAM) of the standard’s objectives thus includes remote failure indication, remote loopback, and link monitoring.

For the physical links, optical transmission offers higher capacities and longer reaches. As there are nevertheless significant regional differences [21], the EFM specification describes 14 different PHYs. These are divided into three main categories: optical Point-to-Point (P2P) for 10 km@100 Mbps/1 Gbps, optical Point-to-Multipoint (P2MP) for 10/20 km@1 Gbps (enhanced to 10 Gbps with IEEE 802.3av in 2009), and electrical (“copper”) P2P for 0.75 km@10 Mbps or 2.7 km@2 Mbps. The P2MP version, also called “Ethernet Passive Optical Network” (EPON), was added for efficiency reasons. Albeit requiring a special adaptation in the (multipoint) MAC Control, it either eliminates the need to lay long individual links per user or the need for a complex switch [68]. In 2012 approximately 17% of all 624 million broadband accesses were optical [69], with EPON at an estimated 40 million nodes the most successful of the available optical technologies [70].

“Carrier Ethernet” extends Ethernet for use as a Wide Area Network (WAN) technology in the core networks of telecommunication service providers. The goal is to enable end-to-end layer 2 Ethernet, which in the core network might go over a variety of underlying technologies like microwave radio, Optical Transport Network, Synchronous Optical NETworking (SONET)/Synchronous Digital Hierarchy (SDH), as well as native Ethernet. To achieve the needed scalability, resiliency, and manageability in the core network a variety of layer 2 principles from IEEE 802.1 and 802.3 can be adopted. Examples are VLANs, prioritization, the OAM defined for EFM, MAC addressing, switching, link aggregation, as well as variants of the Spanning Tree Protocol [66]. Another important concept is label switching. The IETF is publishing specifications for Multi-Protocol Label Switching (MPLS), which allows adding a label to an IP packet with a predefined tunnel, i.e., a predefined path that switches a packet in accordance to the rules of the network technology the packet is going over [66].

The Metro Ethernet Forum (MEF), an organization dedicated to the deployment of Carrier Ethernet, provides a suite of specifications suitable for the implementation of [71] and certification for Carrier Ethernet. “Carrier Ethernet CE 2.0,” the MEF launched at the beginning of 2012 [72], further improves the options available to providers when implementing their Carrier Ethernet by increasing the number of services and their manageability over interconnected provider networks. Market research companies see Carrier Ethernet growing continuously alongside the IP traffic in the network (see, e.g., [73]).

In respect to automotive, the requirements of Carrier Ethernet have little in common with the requirements in cars, where the links are short – even compared with standard LAN Ethernet – and where the traffic is predictable. Nevertheless, when not looking at in-vehicle networking but at the car as an additional node in the networked world, the all-prevailing IP determines the services the car needs to be connected to. Various automotive specific applications in terms of diagnostics, etc. are envisioned and the telecommunications industry has to have the capacity to support these, in most cases wirelessly.

1.2.5 "Automotive Ethernet"

The previous sections have presented the various uses of "Ethernet" in different industries. Every industry has kept some parts of the original "IEEE Ethernet" but has amended or changed others according to the particular needs and the structure of the respective market.

Automotive relates to the other industries in various ways. First, the automotive industry is one of the biggest customers for industrial automation companies [74]. Second, the car is going to be another node in the telecommunication network. While the first is of no concern for this book, the second relates to "Automotive Ethernet" in that seamless connectivity between the car and "the rest of the world" provides an attractive outlook into relevant future use cases. From a technical viewpoint "Automotive Ethernet" is yet something different.

The focus of "Automotive Ethernet" is on in-vehicle networking, i.e., the communication between the various Electronic Control Units (ECUs) inside the car. For this, the automotive industry would like to reuse as much of existing technologies as possible, over all protocol layers. At the same time, the industry sells around 70 million cars a year, with every car containing various ECUs that need to be connected. This means that the expected market volume is big enough to justify the development of a special PHY in order to meet the automotive requirements while reducing the costs to a level that makes the technology attractive for the industry. At the time of writing, the two Automotive Ethernet PHY technologies that had been developed were 100BASE-T1 and 1000BASE-T1 (see also Chapters 3 and 4). While this can justify calling only those PHY technologies "Automotive Ethernet," we think it is not sufficient as Automotive Ethernet entails more.

Figure 1.9 shows the simplified ISO/OSI layer model of "Automotive Ethernet." The descriptions of the use of Ethernet in other industries indicated it: It is almost impossible to limit the explanations to just the PHY and the MAC. "Automotive Ethernet" – or more correctly "Ethernet-based communication in automotive" – covers all layers of the ISO/OSI layering model. A very important motivation for the industry to use Ethernet is that protocols for all levels are available and that the industry can select the appropriate solution for each layer. Additionally, the clearly defined interfaces between the layers even allow the development of new protocols for individual layers while reusing protocols for the rest. It is the goal of this book to explain how each layer is addressed in automotive. Chapter 4 explains aspects related to the Physical Layer, including different PHY variants. Chapter 5 discusses the protocols used on layers 2–7: Section 5.1 for AVB, Section 5.2 for VLANs, Section 5.3 for IP, and Section 5.4 for the application layers, i.e., middleware. But, this book shows also that Automotive Ethernet goes beyond the ISO/OSI layering model. With background and history explained in Chapters 2 and 3, Chapter 6 addresses overall Electric and Electronics (EE) architecture and tooling, while Chapter 7 outlines future developments.

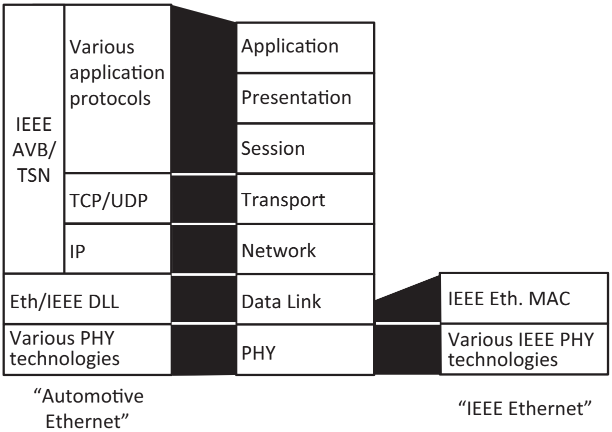

Figure 1.9 Simplified protocol stack of Ethernet-based communication in automotive ("Automotive Ethernet") in comparison.

1.3 Comparison of Markets

Figure 1.10 compares the number of Ethernet ports and the overall market revenue of various industries. Next to the industries discussed in Section 1.2, other industries such as professional audio (see also Section 5.1), architecture, and entertainment lighting are considered. It can be seen that in respect to port usage, there are two groups: those with less than 10 Mio. ports per year and those with more than 100 Mio. ports per year. Naturally, those markets with smaller volumes rather reuse/have to reuse the PHY hardware, as it is more difficult to justify the development of a specific PHY development.[13] They have to make sure that industry specific adaptations can be realized with help of software and special ASICs. For Industrial Automation and Aviation e.g., significant changes are implemented at the MAC layer.

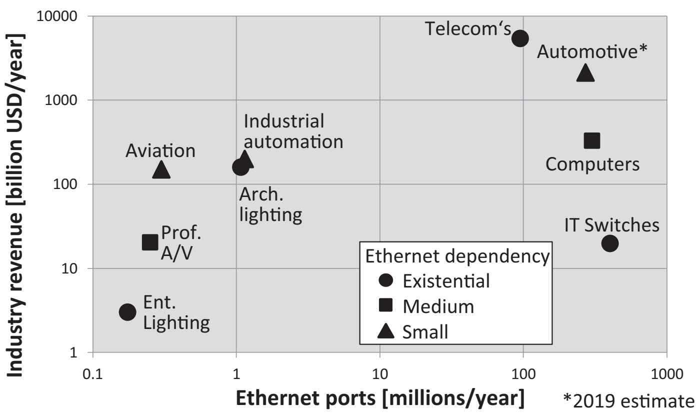

Figure 1.10 Ethernet ports, overall market revenue, and dependency of market on Ethernet for different industries (mainly year 2012 data; see Table 1.3 for references).

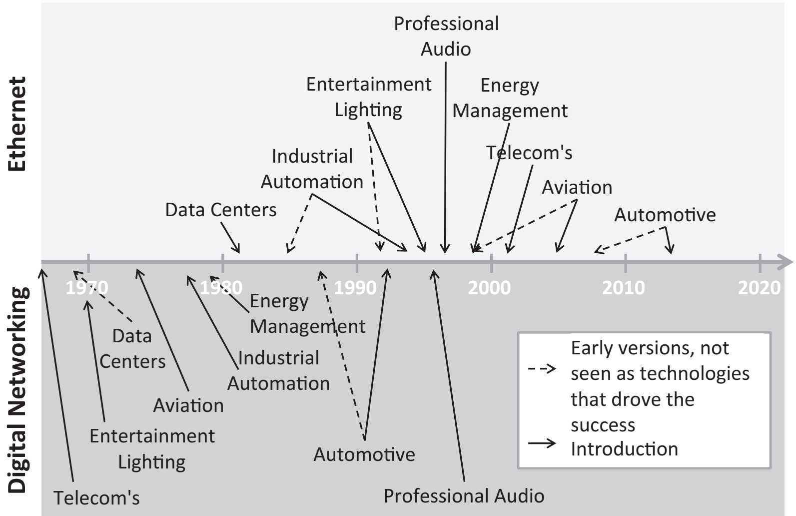

Figure 1.11 Timeline of introducing digital networking versus introducing Ethernet in different industries.

This is different for the markets with large volumes. Telecommunications, e.g., is a prime application area for the optical PHY solutions that have specifically been develop to cater for the long range requirements of the industry. At the same time "Carrier Ethernet" requires specific management functions, which are realized with IEEE 802.1 protocols or others found in the specifications of the Metro Ethernet Forum (MEF). Automotive also justifies the development of specific PHY that meet the specific cost and performance target, while Computers and IT Switches represent the original market for IEEE 802.3 PHY specifications anyway.

Figure 1.11 depicts the different points in time when digital networking was introduced for different industries and when Ethernet was introduced. It can be seen that the automotive Industry is currently the latest industry to introduce Ethernet and that this happened about 20 years after the introduction of digital networking in the car. When comparing this with other industries, timing does not seem to matter. For some other industries like telecommunications or aeronautics this gap is even larger. Other industries, like data centers, thrived directly with Ethernet.

Table 1.3 gives an overview of some of the specific properties of the various Ethernet using industries. As can be seen, the specifics of these industries are quite diverse. The differences start with the type of product for which Ethernet is being used. It can be a consumer good/service or an investment good to produce consumer goods and services. The end-products values can vary between a few hundred to a few millions of dollars. The volume in which it is produced can vary between a few thousand to hundreds of millions per year. Next, the market for the Ethernet parts might be centralized around very few customers or consist of a multitude that sell the end-product. Some are heavily dependent on Ethernet (like switches), for others it is a small part of their product (like

Table 1.3 Main properties of different Ethernet market segments

	Data centers	Consumer products	Telecommunications	Industrial automation	Aviation	Automotive
Implementer[a]	LAN equipment manufacturer	Consumer electronics manufacturers	Communication providers (carriers)	Industrial Automation suppliers	Aircraft manufacturers	Car manufacturers
Product	LAN equipment	(Networked) CE like computers, printers, . . .	Communication services	Industrial automation equipment	Planes (helicopters)	Cars (trucks, busses, utility vehicles)
Customer	All entities/ companies with a LAN	Consumers /companies	Consumers /companies	All manufacturers, building operators	Airlines, Military, . . .	Consumers /companies
Industry structure	Top 1 holds >60%, Top 5 share >80% of Ethernet LAN switches [75]	Top 5 share >50% of the computer[b] market [76], but tablets are taking over [77]	Top 12 share 50% of mobile subscribers [78]; overall 798 mobile network operators [79]	~Top 11 companies share 50% of the market, ~50 share 90% of the market [54]	Top 2 company share >60% (incl. military, duopoly in some segments), 10 manufacturers >90% [80]	Top 5 sell >50% of cars, top 18 sell >90%
Market 2012, revenue in Billion USD (B$) or pieces	[64] 19.8B$ Ethernet LAN switches	[81] Computer sales 329B$ [82] microprocessors for PCs 40B$; (@ 352.7Mio PCs [76])	[83] Telecom service 2200B$; [73] carrier Eth. equip. 34B$; (@ 6.8B mobile, 1.2 B. fixed line subscribers, 750 Mio. Internet households [84])	[74] Indus. automation supplier 200B$; [85] 75B$ of which electronics; [86] 2.6B$ wireline industrial networking	[80] Aircrafts 150B$; [87] avionics 6.3B$ (@ ~1500 planes [88])	[89] Cars 2130B$, supplier 631B$; [90] automotive semicond. 25.5B$; [91] networking semicond. 0.55B$ (@ ~70 Mio. cars)
Year of Ethernet intro	1981 [6]	1981 [6]	2000 [54]	~2000 [31] (1985 [92])	2005 [93] (2000 [46])	2013 (2008; see Chapter 3)
Key Eth. Requirements	Original Ethernet use	Original Ethernet use	Long reach, management, QoS	Short response time and reliability	Real time, reliability, weight	Costs (EMC), weight, data rate
Ethernet market in ports	>400 Mio. (2012) [94]	Rough estimate:[c] 300 Mio. but decreasing	95 Mio. (2017) [73].	~1.14 Mio. (2012) [39]	Rough estimate: 300 T/y	[95] >270 Mio. (2019)

[a] "Implementer" is the one most likely to drive the decisions on the networking technology used. This includes one or the other case in which the implementer's customer makes the choice.
[b] Computers represent just one of a variety of products so that overall there are more vendors.
[c] Most desk-based and mobile PCs' [76] share of DSL routers for homes [69], some printers, game consoles, etc.

in cars or planes). The most astounding observation thus is, that despite the differences in requirements and circumstances, the solution "Ethernet" works for all.

Notes

1 In 1980, Robert Metcalfe, the author of the Xerox internal memo that first mentioned Ethernet, presented what became the so-called Metcalfe's Law. This states that the value of a telecommunications network increases proportionally with the square of the number of compatible communication devices [96]. It can thus be expected that the thinking behind this was used to promote the idea of a networking technology that more than one company can use to build products.

2 During the celebration of 40 years of Ethernet at the IEEE 802 plenary meeting in November 2013 in Dallas, Robert Metcalfe stated that it had been on their lawyers' advice that the DIX group opened up the standard and later offered it to the IEEE. Restricting the technology to DEC, Intel, and Xerox would have violated antitrust laws.

3 The same CSMA/CD mechanism is still used in environments where the channel is always shared, such as in the wireless communication of IEEE 802.11 WLANs (sometimes disconcertingly referred to as "wireless Ethernet"). Some sharing (but not CSMA/CD) was reintroduced with Ethernet in the First Mile (EFM), for which link distances of up to 20 km were defined. In one use case, this included the possibility to share the link without switches, in order to allow for a cost-efficient installation of the cabling.

4 Point-to-Point (P2P) communication primarily refers to a direct communication between two units. Depending on the communication aspect (=ISO/OSI layer) under discussion, this can nevertheless have completely different meanings. Historically, the expression P2P was used for a point-to-point data link with exactly two end points that was not part of a network and on which no data or packet formatting was performed. Generally, this was based on an RS-232 interface or similar [99], which coincidentally is also something the car industry considered in the very early days of in-vehicle networking. At the other end of the spectrum, P2P was and is also used in reference to the end nodes (i.e., only in respect to the higher layers), such as in the one-to-one communication of a telephone call, independent of the physical network layout.

In the switched Ethernet network under discussion, P2P refers to the communication on layers 1 and 2/3, depending on the type of data transported. No matter how many hops there are between two end units or how many end units there are, for each hop, exactly two nodes are physically connected, the data are meant to go to/pass through the direct communication nodes, and the use of the medium is controlled by those two units.

In the context of the in-vehicle networking discussed in this book, P2P is used when there is a direct physical connection between two units only and when those two units are unambiguously identified as the next communication partners so that they do not have to directly share the bandwidth. This is the case in an Automotive Ethernet network but is also the case for a one-to-one communication over, e.g., LVDS/pixel links or USB (see Sections 2.2.6 and 2.2.7). This is clearly different from the case of LIN and CAN, where more than two nodes can be physically connected and the medium is shared. This is also different from MOST or FlexRay, where often only two units are directly physically connected. Nevertheless, this is on layer 1 only. On layer 2, the link between those units is also shared among all participants of the network.

5 In the context of IEEE, "half-duplex operation" is often synonymously used for CSMA/CD Ethernet operation, while "full duplex" stands for the switched Ethernet network. In the sense that full duplex refers to a communication in which transmission and reception can happen at the same time, while for half-duplex, either transmission or reception is possible, this is,

of course, correct. In automotive, this reference would nevertheless be unusual. Instead, the CSMA/CD system would be called a “bus” system (meaning the medium is shared among several users), while the switched Ethernet operation would be referred to as “switched.” At the beginning, the automotive industry referred to it also as “Point-to-Point” (P2P) (the medium connects two users only, and only those two units decide on the loading of the channel). However, P2P is also not unambiguous, so “switched” is the most appropriate term from an automotive perspective. This identifies directly what is relevant.

It is, e.g., ambiguous, to the authors, to call switched 100BASE-TX Ethernet “full duplex.” In case of 100BASE-TX, one wire pair is used for transmitting and one wire pair for receiving. This means it is “half-duplex” in the sense that transmission and reception do not happen simultaneously on the same wire pair, like we would expect for “full duplex.” At the same time, the bandwidth in 100BASE-TX is not shared, as with Ethernet CDMA/CD or other bus systems. The communication still happens in a switched network, which is what is relevant from IEEE for labeling the connection “full duplex.” “Switched” communication is thus the least ambiguous term for this type of communication. In the context of this book, “half-duplex” will *not* be a direct placeholder for Ethernet in CSMA/CD mode but for cases in which the incapability to simultaneously receive and transmit on PHY level on the same medium is relevant.

6 Not only Ethernet interfaces receive MAC addresses; they are also received by the interfaces of other IEEE communication technologies and of technologies standardized in other organizations. Examples of the latter are FDDI and ATM. The MAC address originates from the original Xerox Ethernet addressing scheme. Today, MAC address assignments are coordinated by IEEE-RA [100].

7 Apparently, this is also a legacy from CSMA/CD operation, where packets needed to be long enough in order to process a collision.

8 It seems that the name “fieldbus” was used a while, before meaning and definitions were added to it [29]. Most simplified, a fieldbus connects units “in the field,” i.e., units distributed on a factory floor, which can have dimensions large enough to be fieldlike.

9 Industrial automation represents a safety critical environment with challenges different from what can be found in data centers. So, when the computerization and local networking reached the automation industry, other aspects needed to be standardized additional to the (LAN/fieldbus) communication technology. The programming of the PLCs as such needed to be standardized (IEC 61131). The installation of the communication networks was (and still is) a challenge (IEC 61918). Not only is the environment not always friendly in respect to temperature, vibration, dirt, acids, etc., it is also necessary to keep wiring changes to a minimum when the line is changed in order to produce a different product. Other topics like redundancy (e.g., IEC 62439), the development of safety critical systems as such (e.g., IEC 61508), functional safety (e.g., IEC 61511), security for industrial automation and control systems (e.g., IEC 62443), parameterization, and diagnosis are also important.

10 “Industrial Ethernet” is an expression so commonly used that it seems to defy the need for an unambiguous definition. It generally refers to the case when some or even all elements of Ethernet as defined in IEEE 802 are (re)used in an industrial environment for tasks directly related to the manufacturing process fulfilling exactly those additional requirements for robustness, high availability (e.g., ring redundancy), functional safety, cyber security, etc. The expression was apparently introduced with “Profinet” [97].

11 It was, e.g., a major concern for the automotive industry when standardizing FlexRay that a reuse in aviation could lead to liability issues for the car industry.

12 The extension “Carrier” originates in the jargon for telecommunications network providers: “Common Carriers.” “Carrier Ethernet” as such is more the name for a market segment than one specific technology.

13 The intention to standardize a 10 Mbps single pair long reach Ethernet PHY version for Industrial Automation at IEEE 802.3. Reference [41] somewhat contradicts the statement that for

markets with small volumes it is not worthwhile to develop a separate PHY. The initiators of this PHY acted on the assumption that with this solution the market will increase significantly to what is depicted in Figure 1.10, owing to the possibility to better compete with fieldbus technologies that today still make about 2/3 of all newly installed nodes [98]. However, even a threefold market for Industrial Ethernet is still small in comparison to markets of other industries. But in the end, it is up to the silicon vendors to decide on the market opportunities.

References

[1] M. Lasar, "The UNIX Revolution – Thank You, Uncle Sam?," arstechnica, 19 July 2011. [Online]. Available: http://arstechnica.com/tech-policy/2011/07/should-we-thank-for-feds-for-the-success-of-unix/. [Accessed 23 May 2013].

[2] B. Montague, "Why You Should Use a BSD Style License for Your Open Source Project," freebsd, 17 May 2013. [Online]. Available: www.freebsd.org/doc/en/articles/bsdl-gpl/index.html. [Accessed 21 May 2013].

[3] Wikipedia, "Berkeley Software Distribution," Wikipedia, 28 July 2016. [Online]. Available: http://en.wikipedia.org/wiki/Berkeley_Software_Distribution. [Accessed 31 July 2016].

[4] V. G. Cerf, "Final Report of the Stanford TCP Project," 1 April 1980. [Online]. Available: ftp://ftp.rfc-editor.org/in-notes/ien/ien151.txt. [Accessed 5 July 2013].

[5] Wikipedia, "Internet Protocol Suite," Wikipedia, 21 July 2016. [Online]. Available: http://en.wikipedia.org/wiki/Internet_protocol_suite. [Accessed 31 July 2016].

[6] U. v. Burg and M. Kenny, "Sponsors, Communities, and Standards: Ethernet vs. Token Ring in the Local Area Networking Business," *Industry and Innovation*, vol. 10, no. 4, pp. 351–375, December 2003.

[7] R. M. Metcalfe, D. R. Boggs, C. P. Thacker, and B. W. Lampson, "Multipoint Data Communication System (with Collision Detection)," U.S. Patent 4,063,220, 31 March 1975.

[8] D. Boggs and R. Metcalfe, "Ethernet: Distributed Packet Switching for Local Computer Networks," *Communications of the ACM*, vol. 19, no. 7, pp. 395–405, July 1976.

[9] S. Carlson, "Fast, Simple, Reliable and Cheap: Ethernet at 40," September 2013. [Online]. Available: https://standards.ieee.org/events/automotive/00_Carlson_keynote_techday3_0913.pdf. [Accessed 31 July 2016].

[10] Digital Equipment Corporation, Intel Corporation, Xerox Corporation, "The Ethernet, A Local Area Network. Data Link Layer and Physical Layer Specifications, Version 1.0," 30 September 1980. [Online]. Available: http://ethernethistory.typepad.com/papers/EthernetSpec.pdf. [Accessed 23 June 2013].

[11] C. E. Surgeon, *Ethernet: The Definite Guide*, Sebastopol, CA: O´Reilly, 2000, February.

[12] Wikipedia, "IEEE 802.3," Wikipedia, 30 July 2016. [Online]. Available: http://en.wikipedia.org/wiki/IEEE_802.3. [Accessed 31 July 2016].

[13] AT&T, "Milestones in AT&T History," [Online]. Available: www.corp.att.com/history/milestones.html. [Accessed 23 June 2013].

[14] R. M. Metcalfe, "The History of Ethernet," www.youtube.com/watch?v=g5MezxMcRmk, 2006 [Accessed 26 February 2017].

[15] R. Braden, "RFC 1122: Requirements for Internet Hosts – Communication Layers," October 1989. [Online]. Available: http://tools.ietf.org/pdf/rfc1122.pdf. [Accessed 23 June 2013].

[16] R. Braden, "RFC 1123: Requirements for Internet Hosts – Application and Support," October 1989. [Online]. Available: tools.ietf.org/pdf/rfc1123.pdf. [Accessed 26 June 2013].

[17] F. Baker, "RFC 1812: Requirements for IP Version 4 Routers," June 1995. [Online]. Available: http://tools.ietf.org/pdf/rfc1812.pdf. [Accessed 23 June 2013].

[18] Microsoft, "Verfügbarkeit von Microsoft Windows 95 Service Pack 1-Komponenten," 14 February 2006. [Online]. Available: https://support.microsoft.com/de-de/kb/142794. [Accessed 16 November 2016].

[19] L. Völker, *Modified from nonpublic document*, 2011.

[20] IEEE 802 LMSC, "Agenda & Minutes (Unconfirmed) – IEEE 802 LMSC Executive Committee Meeting (updated 24 September 2007)," 20 July 2007. [Online]. Available: www.ieee802.org/minutes/jul2007/Minutes%20-%20Friday%20July%2020%202007.pdf. [Accessed 26 June 2016].

[21] W. W. Diab and H. M. Frazier, *Ethernet in the First Mile*, New York: IEEE, 2006.

[22] IEEE 802, "IEEE 802 LAN/MAN Standards Committee," 2016. [Online]. Available: www.ieee802.org/. [Accessed 29 June 2016].

[23] IEEE Computer Society, *IEEE Std 802.1Q 2011, Media Access Control (MAC) Bridges and Virtual Bridge Local Area Networks*, New York: IEEE, 2011.

[24] IEEE, "Ethertype," constantly updated. [Online]. Available: http://standards.ieee.org/develop/regauth/ethertype/eth.txt. [Accessed 31 July 2016].

[25] W. Lawrenz, *CAN Controller Area Network Grundlagen und Praxis*, 2nd edition, Heidelberg: Hürthig, 1997.

[26] L. Larsson, "Fourteen Industrial Ethernet Solutions under the Spotlight," September 2005. [Online]. Available: www.iebmedia.com/index.php?id=4811&parentid=74&themeid=255&hft=28&showdetail=true&bb=1&PHPSESSID=38a2saj9cvll4kaguj5m14c4j5. [Accessed 24 July 2013].

[27] M. Felser and T. Sauter, "Standardization of Industrial Ethernet – the Next Battlefield?," *Proceedings of the 5th IEEE International Workshop on Factory Communication Systems*, Vienna, 22 September 2004.

[28] S. Djiev, "Industrial Networks for Communication and Control," [Online]. Available: http://anp.tu-sofia.bg/djiev/PDF%20files/Industrial%20Networks.pdf. [Accessed 25 July 2013].

[29] T. Sauter and M. Wollschlaeger, "Feldbussystems – Historie, Eigenschaften und Entwicklungstrends," *Informationstechnik und Technische Informatik*, vol. 42, no. 4, pp. 7–16, 2000.

[30] G. Färber, "Feldbus-Technik heute und morgen," *Automatisierungstechnische Praxis*, no. 11, pp. 16–36, 1994.

[31] M. Felser and T. Sauter, "The Fieldbus War: History or Short Break Between Battles?," *4th IEEE International Workshop on Factory Communication Systems*, 28 August 2002.

[32] IMS Research, "How Sustainable is an Industrial Fieldbus Infrastructure?," Automation World, 28 February 2013. [Online]. Available: www.automationworld.com/networking-amp-connectivity/how-sustainable-industrial-fieldbus-infrastructure. [Accessed 28 July 2013].

[33] Contemporary Controls, "Dick Caro – One of the Most Influential Persons in the Field of Industrial Networking – Part 2," 2006. [Online]. Available: www.ccontrols.com/pdf/Extv6n3.pdf. [Accessed 15 August 2013].

[34] IEEE 802, "Re: [802SEC] Disbanding of 802.4 and 802.9," *IEEE* 802, 30 July 2004. [Online]. Available: http://ieee802.org/secmail/msg05489.html. [Accessed 27 July 2013].

[35] Schneider Electric, "Chapter 9: Industrial Networks," [Online]. Available: www.schneider-electric.hu/documents/automation-and-control/asg-9-industrial-networks.pdf. [Accessed 22 July 2013].

[36] B. Vogel-Heuser et al., "FuRIOS: Fieldbus and Remote I/O System Comparison," Pepperl&Fuchs GmbH, Mannheim, 2002.

[37] J.-P. Thomesse, "Fieldbus: Industrial Network Real-time Network," 2005. [Online]. Available: http://etr05.loria.fr/slides/vendredi/ETR2005_Thomesse.pdf. [Accessed 22 July 2013].

[38] Laboratory of Process Data Processing of Reutlingen University, "Information about Real-Time Ethernet in Industry Automation," Laboratory of Process Data Processing of Reutlingen University, 2008. [Online]. Available: www.real-time-ethernet.de/. [Accessed 12 July 2013].

[39] Industrial Ethernet Book, "The World Market of Industrial Ethernet," April 2012. [Online]. Available: www.iebmedia.com/index.php?id=8595&parentid=74&themeid=255&hpid=2&showdetail=true&bb=1&appsw=1&sstr=world market. [Accessed 27 July 2013].

[40] IEEE-SA, "IEEE Standard for Ethernet: Amendment: Specification and Management Parameters for Interspersing Express Traffic," IEEE-SA, New York, 2016.

[41] L. Winkel, M. McCarthy, D. Brandt and K. Matheus, "10Mb/s Single Twisted Pair Ethernet Call for Interest," 26 July 2016. [Online]. Available: www.ieee802.org/3/cfi/0716_1/CFI_01_0716.pdf.

[42] Avionics Interface Technologies, "ARINC 429 Protocol Tutorial," Avionics Interface Technologies, undated. [Online]. Available: http://aviftech.com/files/2213/6387/8354/ARINC429_Tutorial.pdf. [Accessed 14 July 2013].

[43] J. Wittmer, "ARINC 429 Ein Avionik Feldbus der zivilen Luftfahrt," Berner Fachhochschule, 23 January 2007. [Online]. Available: https://prof.hti.bfh.ch/uploads/media/ARINC_429.pdf. [Accessed 14 July 2013].

[44] Condor Engineering, "AFDX Protocol Tutorial," May 2005. [Online]. Available: www.cems.uwe.ac.uk/~ngunton/afdx_detailed.pdf. [Accessed 15 July 2013].

[45] C. B. Watkins and R. Walters, "Transitioning from Federated Avionics Architectures to Integrated Modular Avionics," *Trans. of Digital Avionics Systems Conference*, 21 October 2007.

[46] J. W. Ramsey, "Boeing's 767–400ER: Ethernet Technology Takes Wing," 1 May 2000. [Online]. Available: www.aviationtoday.com/av/commercial/Boeings-767-400ER-Ethernet-Technology-Takes-Wing_12652.html. [Accessed 2 August 2013].

[47] L. Buckwalker, Avionics Databusses, 3 ed., Leesburg: http://avionics.com, 2005.

[48] AIM GmbH, "AFDX Workshop," March 2010. [Online]. Available: www.afdx.com/pdf/AFDX_Training_October_2010_Full.pdf. [Accessed 13 July 2013].

[49] RTCA, "Software Considerations in Airborne Systems and Equipment Certification, DO-178C," RTCA, Washington, 2011.

[50] Joint Aviation Administration, "Advisory Material Joint – AMJ 25.1309," 1998. [Online]. Available: www.mp.haw-hamburg.de/pers/Scholz/materialAFS/AMJ25.1309-Change15.pdf. [Accessed 14 September 2013].

[51] Wikipedia, "Avionics Full-Duplex Switched Ethernet," Wikipedia, 30 June 2016. [Online]. Available: http://en.wikipedia.org/wiki/AFDX. [Accessed 23 July 2016].

[52] Wikipedia, “History of the Telephone,” 23 July 2016. [Online]. Available: http://en.wikipedia.org/wiki/History_of_the_telephone. [Accessed 31 July 2016].
[53] US Department of Transportation, “Telecommunications Handbook for Transportation Professionals: The Basics of Telecommunications: Chapter 1. Telecommunication Basics,” 3 January 2005. [Online]. Available: www.ops.fhwa.dot.gov/publications/telecomm_handbook/chapter1.htm. [Accessed 8 August 2013].
[54] S. Cherry, “A Washington, D.C. Committee Wants to Phase Out Traditional Telephony Worldwide, by June 2018,” 19 December 2012. [Online]. Available: http://spectrum.ieee.org/podcast/telecom/internet/the-end-of-the-public-phone-network. [Accessed 8 August 2013].
[55] Wikipedia, “Telefonnetz,” 1 May 2016. [Online]. Available: http://de.wikipedia.org/wiki/Telefonnetz. [Accessed 31 July 2016].
[56] J. J. Río, A. v. Maltzahn and J. Ong, “The Future of Telecoms: New Models for a New Industry,” Feburary 2012. [Online]. Available: www.leslivresblancs.fr/commerce-et-economie/filieres-specialisees/tic/livre-blanc/the-future-of-telecoms-new-models-for-a-new-industry-1966.html. [Accessed 31 July 2016].
[57] J. Armstrong, “A History of Erlang,” 21 May 2009. [Online]. Available: http://web.archive.org/web/20090521100504/www.cs.chalmers.se/Cs/Grundutb/Kurser/ppxt/HT2007/general/languages/armstrong-erlang_history.pdf. [Accessed 7 August 2013].
[58] Wikipedia, “EWSD,” 31 May 2016. [Online]. Available: http://en.wikipedia.org/wiki/EWSD. [Accessed 31 July 2016].
[59] Information Sciences Institute University of Southern California, “Internet Protocol,” September 1981. [Online]. Available: http://tools.ietf.org/html/rfc791. [Accessed 4 August 2013].
[60] Information Sciences Institute University of Southern California, “Transmission Control Protocol,” September 1981. [Online]. Available: http://tools.ietf.org/html/rfc793. [Accessed 4 August 2013].
[61] P. Barford, “The World Wide Web,” University of Wisconsin, 11 September 2008. [Online]. Available: http://pages.cs.wisc.edu/~pb/640/web.ppt. [Accessed 6 August 2013].
[62] iLocus, “The 10 That Established VoIP,” September 2007. [Online]. Available: www.ilocus.com/sites/default/files/The%2010%20That%20Established%20VoIP.pdf. [Accessed 8 August 2013. no longer available, see https://de.scribd.com/document/275142588/The-10-That-Established-VoIP].
[63] B. Injaz, “VoIP, how does it work,” 2013. [Online]. Available: http://de.slideshare.net/badr212/how-does-voip-work. [Accessed 9 August 2013].
[64] Wikipedia, “Next Generation Network,” 1 May 2015. [Online]. Available: http://de.wikipedia.org/wiki/Next_Generation_Network. [Accessed 31 July 2016].
[65] US Department of Transportation, “Telecommunications Handbook for Transportation Professionals: The Basics of Telecommunications: Chapter 10. The Future,” 2005. [Online]. Available: www.ops.fhwa.dot.gov/publications/telecomm_handbook/chapter10.htm. [Accessed 8 August 2013].
[66] C. G. Gruber and A. Autenrieth, “Carrier Ethernet Transport in Metro and Core Networks,” 28 September 2008. [Online]. Available: www.netmanias.com/ko/post/cshare/5063. [Accessed 31 July 2016].
[67] M. Ghodrat, “Carrier Ethernet Improves IP Service Delivery,” October 2008. [Online]. Available: www.lightwaveonline.com/articles/print/volume-25/issue-10/technology/carrier-ethernet-improves-ip-service-delivery-54889962.html. [Accessed 31 July 2016].

[68] G. Pesavento, "Ethernet in the First Mile; Point to Multipoint; Ethernet Passive Optical Network Tutorial," 9 July 2001. [Online]. Available: www.ieee802.org/3/efm/public/jul01/tutorial/pesavento_1_0701.pdf. [Accessed 11 August 2013].

[69] POINT topic, "World Broadband Statistics," June 2012. [Online]. Available: http://point-topic.com/wp-content/uploads/2013/02/Sample-Report-Global-Broadband-Statistics-Q2-2012.pdf. [Accessed 11 August 2013].

[70] Wikipedia, "Passive Optical Network," 21 July 2016. [Online]. Available: http://en.wikipedia.org/wiki/Passive_optical_network. [Accessed 31 July 2016].

[71] Metro Ethernet Forum, "MEF Technical Specifications," Metro Ethernet Forum, Continuously updated. [Online]. Available: www.mef.net/carrier-ethernet/technical-specifications. [Accessed 31 July 2016].

[72] P. Marwan, "Carrier Ethernet 2.0 ist an den Start gegangen," 28 February 2012. [Online]. Available: www.zdnet.de/41560473/carrier-ethernet-2–0-ist-an-den-start-gegangen/. [Accessed 11 August 2013].

[73] Infonetics, "Carrier Ethernet market to Top $39B by 2017," 10 May 2013. [Online]. Available: www.infonetics.com/pr/2013/Carrier-Ethernet-Market-Highlights.asp. [Accessed 11 August 2013].

[74] Credit Suisse, "Global Industrial Automation," 14 August 2012. [Online]. Available: https://doc.research-and-analytics.csfb.com/docView?language=ENG&source=emfromsendlink&format=PDF&document_id=994715241&extdocid=994715241_1_eng_pdf&serialid=hDabUewpvOqQcRiLxK7rxIQJZZ8TPLDrYHs47S97OOI%3d. [Accessed 31 July 2013].

[75] IDC, "IDC's Worldwide Quarterly Ethernet Switch and Router Tracker Shows Sequential Improvement in Both Markets," 1 March 2013. [Online]. Available: www.idc.com/getdoc.jsp?containerId=prUS23974313. [Accessed 9 September 2013, no longer available on original site; see www.businesswire.com/news/home/20130301005268/en/IDCs-Worldwide-Quarterly-Ethernet-Switch-Router-Tracker].

[76] Gartner, "Gartner Says Declining Worldwide PC Shipments in Fourth Quarter of 2012 Signal Structural Shift of PC Market," 14 January 2013. [Online]. Available: www.gartner.com/newsroom/id/2301715. [Accessed 9 September 2013].

[77] NPD Displaysearch, "Tablet PC Market Forecast to Surpass Notebooks in 2013," prweb, 7 January 2013. [Online]. Available: www.prweb.com/releases/The_NPD_Group/NPD_DisplaySearch/prweb10295949.htm. [Accessed 31 July 2016].

[78] Wikipedia, "List of Mobile Network Operators," 13 August 2013. [Online]. Available: http://en.wikipedia.org/wiki/List_of_mobile_network_operators. [Accessed 14 August 2013].

[79] Telecoms Networks, "MNO Directorio 2013 now available," Telecoms Networks, 2013. [Online]. Available: www.telecomsnetworks.com/. [Accessed 14 August 2013].

[80] JEC Composites, "Aircraft Market Forecast," 19 April 2011. [Online]. Available: www.jeccomposites.com/news/composites-news/aircraft-market-forecasts. [Accessed 15 August 2013].

[81] Statistic Brain, "Computer Sales Statistics," 24 August 2012. [Online]. Available: www.statisticbrain.com/computer-sales-statistics/. [Accessed 10 September 2013, update available].

[82] IDC, "Worldwide PC Microprocessor Revenues in 2013 to Rise 1.6% Compared to 2012," TechPowerUp, 16 January 2013. [Online]. Available: www.techpowerup.com/178838/worldwide-pc-microprocessor-revenues-in-2013-to-rise-1-6-compared-to-2012. [Accessed 31 July 2016].

[83] Send2Press Newswire, "Worldwide Telecommunications Industry Revenue to Reach $2.2 Trillion in 2013, says Insight Research Corp.," 28 Jan 2013. [Online]. Available: www.send2press.com/newswire/Worldwide-Telecommunications-Industry-Revenue-to-Reach-2–2-Trillion-in-2013-says-Insight-Research-Corp_2013-01-0128-002.shtml. [Accessed 13 August 2013].

[84] ITU, "ICT Facts & Figures," 2013. [Online]. Available: www.itu.int/en/ITU-D/Statistics/Documents/facts/ICTFactsFigures2013-e.pdf. [Accessed 12 August 2013].

[85] IMS Research, "Industrial Automation," 2013. [Online]. Available: www.imsresearch.com/research-area/Factory_Automation. [Accessed 31 July 2013, no longer available].

[86] T. Shea, "EtherNet/IP Leads the Way in Industrial Automation Connectivity," 28 December 2011. [Online]. Available: http://blog.vdcresearch.com/industrial_automation/2011/12/ethernetip-leads-the-way-in-industrial-automation-connectivity.html. [Accessed 31 July 2013].

[87] M. Phelps, "AEA Report Illustrates Massive Size of Avionics Market," 6 June 2013. [Online]. Available: www.flyingmag.com/news/aea-report-illustrates-massive-size-avionics-market. [Accessed 15 August 2013].

[88] Airbus, "Global Market Forecast 2012–2031," 2012. [Online]. Available: www.airbusgroup.com/dam/assets/airbusgroup/int/en/investor-relations/documents/2012/presentations/2012–31-Global-Market-Forecast/Global%20Market%20Forecast%202012–2031.pdf. [Accessed 31 July 2016].

[89] IMAP, "Automotive and Components Global Report 2010," 2010. [Online]. Available: www.proman.fi/sites/default/files/Automotive%20%26%20component%20global%20report%202010_0.pdf. [Accessed 31 July 2016].

[90] Semicast, "OE Automotive Semiconductor Market Grew Twelve Percent in 2012," 11 March 2013. [Online]. Available: http://semicast.net/wp-content/uploads/2013/07/110313.pdf. [Accessed 14 August 2013].

[91] IHS, "Mobile Video Streaming Drives Demand for Networking Semiconductors in Cars," 3 April 2013. [Online]. Available: http://press.ihs.com/press-release/design-supply-chain/mobile-video-streaming-drives-demand-networking-semiconductors-car. [Accessed 15 August 2013].

[92] A. Zankl, "Zur Geschichte der Automatisierungstechnik," 6 October 2009. [Online]. Available: www.vdi.de/fileadmin/vdi_de/redakteur/bvs/bv_karlsruhe_dateien/GeschichteAutomation.pdf. [Accessed 31 July 2016].

[93] AIM GmbH, "What Is AFDX," 2016. [Online]. Available: www.afdx.com/. [Accessed 31 July 2016].

[94] Ethernet Alliance, "Ethernet Alliance Panel #2: Bandwidth Growth and the Next Speed of Ethernet," 2012. [Online]. Available: www.ethernetalliance.org/wp-content/uploads/2012/04/Ethernetnet-Alliance-ECOC-2012-Panel-2.pdf. [Accessed 15 March 2013].

[95] IEEE 802.3, "Reduced Twisted Pair Gigabit Ethernet Call for Interest," March 2012. [Online]. Available: www.ieee802.org/3/RTPGE/public/mar12/CFI_01_0312.pdf. [Accessed 15 March 2012].

[96] S. Simeonov, "Metcalfe's Law: More Misunderstood Than Wrong?," High Contrast, 26 July 2006. [Online]. Available: http://blog.simeonov.com/2006/07/26/metcalfes-law-more-misunderstood-than-wrong/. [Accessed 5 July 2013].

[97] L. Winkel, *E-mail correspondence*, 2013.

[98] T. Carlsson, "Industrial Network Shares According to HMS," 22 January 2015. [Online]. Available: www.anybus.com/readnews.asp?NID=177. [Accessed 29 June 2016].

[99] Wikipedia, “Point-to-Point (Telecommunications),” Wikipedia, 2 April 2016. [Online]. Available: https://en.wikipedia.org/wiki/Point-to-point_(telecommunications). [Accessed 31 July 2016].

[100] Wikipedia, “MAC Address,” Wikipedia, 14 July 2016. [Online]. Available: http://en.wikipedia.org/wiki/Mac_address. [Accessed 31 July 2016].

www.cargeek.ir

2 A Brief History of In-Vehicle Networking

2.1 Role of In-Vehicle Networking

An explanation of the needs, the development, and some of the choices in in-vehicle networking starts with the windows. When automobiles were invented they were simply machines on wheels without windows. It was only later that windows were added, first at the front, then at the sides and back. The windows were static or insertable in one piece. This obviously was not very comfortable, neither for the handling of the windows, nor for the temperature regulation in the passenger cabin. Thus, in 1928 the first mechanical window winder, able to hold a window at any position desired, was presented to the public [1]. The first power windows were introduced in 1941 [2]. BMW was the first company to introduce power windows in Europe and the first BMW with all electric power windows was a “Series 2 BMW 503,” which had an SOP at the end of 1957 [3]. This is where it gets interesting.

It is quite straightforward to imagine a switch in a vehicle door that actuates the electric motor for a window located in the same door. Everything is in one physical location and the wiring will be short. The wiring gets longer when all movable windows are required to be controllable by the driver, in addition to the “local” control in every door. More wiring between almost exactly the same locations is needed if a central door lock with discrete wiring is added, even more with an additional electronic side mirror adjustment. Figure 2.1(a) gives an idea that with only the basic comfort functions the size, weight, and number of wires will soon become prohibitive. In the case of discrete wiring, inventiveness quickly circles around the question of “How is it possible to fit another wire onto this inline connector or through this opening between, e.g., body and door?” instead of fully exploring the possibilities of creating a new feature. On top of this, large wiring bundles are not only heavy, costly, and hard to install, but also error prone and difficult to diagnose [4].

Figure 2.1(b) shows the same communication structure – the logic stays in every door – realized with a bus system. The amount of cabling and the associated weight are significantly reduced. In the diagram the door-to-door communication is shown as a standalone system. This is not necessarily the case. A bus might easily be connected to other functions in the car. Examples of new combination functions could be to activate the light when the car is being unlocked, to remotely open a window if the key has been locked inside, or to automatically close the windows if it rains. Figure 2.1(c) shows a different option when using in-vehicle networking. The intelligence is concentrated

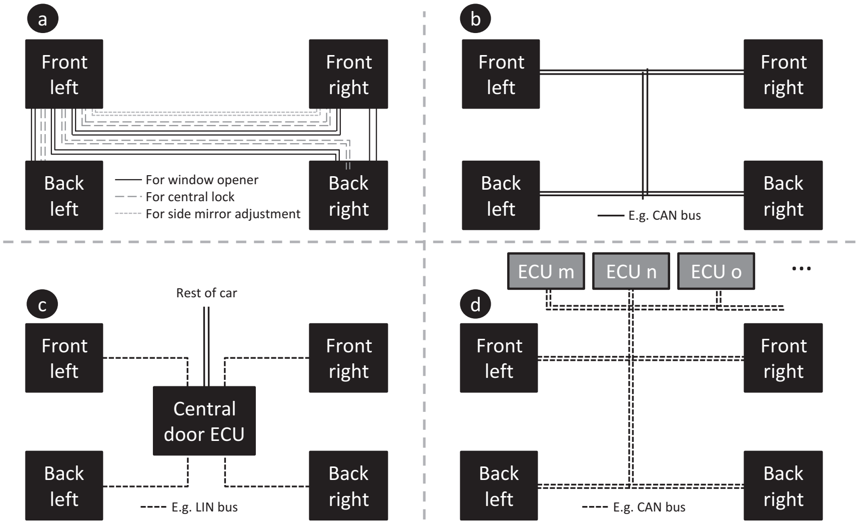

Figure 2.1 Wiring options for electric window control, central look, and side mirror adjustments (see also [4]).

in one Electronic Control Unit (ECU) that controls all door functions. This potentially allows the complexity of the features provided to be increased. An efficient solution might also be found with a (not depicted) mixture of (b) and (c), in which a central ECU processes the information that is the same for all doors or car models, whereas smaller door ECUs handle door- or car-specific functions.

As a next step, the in-vehicle networking is an enabler for an Electrics and Electronics (EE) architecture in which the software is distributed. It can be realized such that there is no distinct ECU for the door control but the processing power of idle ECUs is used instead (see Figure 2.1(d)). Considering the little amount of time that some functions are used in the car – those related to the door provide good examples – this potentially reduces costs and/or frees processing resources that can then again be used for other innovations.

The simple example just described gives an idea of the complexity involved. There are various other choices in relation to the architecture and various more criteria to consider. Depending on the car model envisioned and technologies available a car manufacturer will enable and partition functionalities in respect to costs, weight, number of ECUs, installation effort, harness diameter, available communication bandwidth, sales prognosis, and more (see also [4] or Chapter 6 for the specific impact Automotive Ethernet has on the architectural choices). The important point is that innovativeness and technical possibilities form a virtuous circle. Because an inventor wants to realize a certain function, he/she pushes on the bounds of the technical resources. The availability of technical resources encourages innovators to make use of them until they reach their limits and push them out again.

Another important aspect to consider refers to the type of data transmitted. In the window example described above the information content of most of the messages is short:

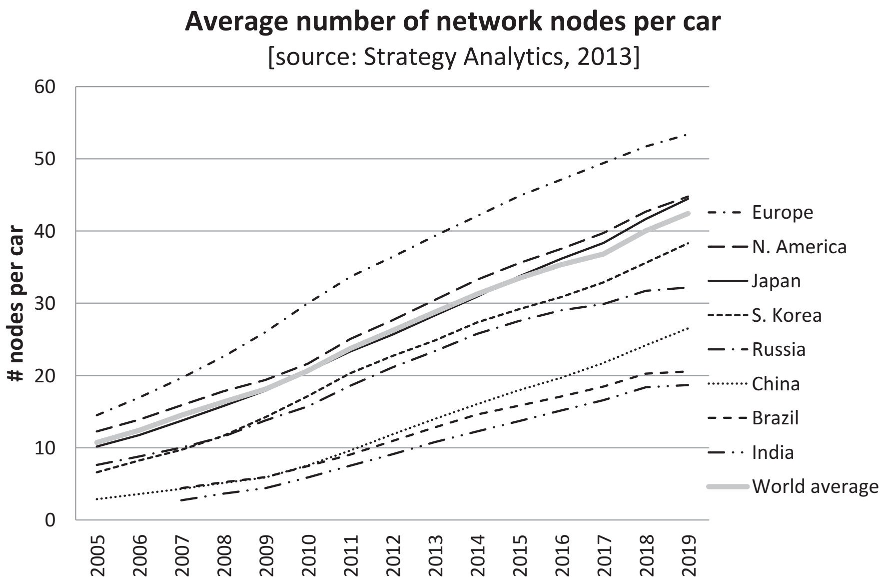

Figure 2.2 Average number of networked nodes per car, depending on region.

on/off, open/close, switch is being activated/switch is no longer activated, window is open/window is closed, velocity of the window movement, etc. The use of a networking technology can save weight, space, and costs. Nevertheless, the communication mechanisms behind it all can be kept simple.

This changes in the case of more complex electronics. For example, if the engine does not function as expected, it needs to be possible to receive differentiated messages from the engine control on the status of the system. The communication system needs protocols to allow it to distinguish between, e.g., special diagnostic messages, standard control messages, regular status messages, and software updates. Once all this information is available on an in-vehicle networking system it can be reused in other units inside the car to, e.g., display "ok" or a warning to the driver. Then some type of message classification/middleware, more sophisticated addressing, and channel use concepts might be required. This in return needs yet more intelligence and refinement with the in-vehicle networking technology.

Figure 2.2 shows the development of the average number of networked nodes in cars for different regions. As can be seen, the number is continuously increasing; every car produced in 2019 is expected to contain on average 42 ECUs that need to communicate! Ever more new functions are realized by electronics and ever more mechanical functions are being replaced by electronics. According to [5], it is expected that regular cars will drive fully autonomously on the road by 2020. In these cars the driver function will be taken over by electronics that again will need to be connected very reliably to the in-vehicle network. Thus, not only the "window example" shows that automotive in-vehicle networking technologies are a fundamental technical resource. The more flexible and scalable the in-vehicle networking technology is, the better it provides a reliable resource for the innovations needed for ever more sophisticated customer functions.

2.2 Traditional In-Vehicle Networking

Section 2.1 motivated the principal need for in-vehicle networking technologies. This section describes the actual technical developments in the automotive industry.

Each of the traditional in-vehicle networking technologies is different in respect to its characteristics. To those who have worked in the field for some time they have become almost like old friends whose strengths and weaknesses are sometimes more and sometimes less enjoyable to deal with. None of the systems is ideal, but each is particularly suitable for a specific use case or physical location inside the car.

The following subsections discuss the early days of in-vehicle networking as well as the in our view most important existing in-vehicle networking technologies: CAN, LIN, FlexRay, and MOST. Additionally, the use of "pixel" and "consumer links" is described. The use cases, technological features, strengths, and limits of the technologies described represent only one side of the coin though. As is the case for almost all technical developments, additional motives to use a certain technology depend on urgency, economics, and politics. Furthermore, choices depend on the way of thinking, the capabilities, and preferences of the individuals working on a solution. The following explanations are intended to help the reader understand where automotive in-vehicle networking is coming from and where the industry is heading. It also helps to understand how necessary but also how radical the changes are that Ethernet brings.

2.2.1 The Early Days of In-Vehicle Networking

The first electronic cables inside cars were dedicated wires between sensors and actuators. With more electronics this became a headache in production, in finding space, and for both reliability and troubleshooting (see also Section 2.1). To decrease the number of wires, first a serial interface was needed and then some type of distribution/addressing mechanism so that several units were able to reuse the same wire and the same information. All car manufacturers had the same issues to solve. Nevertheless, they thought that their capability to handle these issues, i.e., their in-vehicle networking technology, was a differentiating feature [6]. In consequence, the industry started with a variety of car manufacturer specific solutions that, not surprisingly, appeared around the same time that Electric and Electronics (EE or E/E) engineering departments were established. At BMW, this was at the end of the 1980s/beginning of the 1990s [7].

Thus, BMW introduced the first car with a communication bus in 1987. The use case was the diagnosis of the engine control unit and thus called "D-Bus" (D for "Diagnose," English: diagnosis). The communication method used was based on "K-Line," a single-ended, i.e., 1-wire bus for asynchronous data up to 10.4 kbps. K-Line was later standardized as ISO 9141 and is similar to RS-232 [8] [9]. As engine control information was now available digitally it was desired to reuse the data for information to the

Table 2.1 Principal choices to make when designing a networking technology

Challenges for a serial communication technology		
Multiple access	Data rate	Robustness
• Arbitration (message priority based) • Carrier sensing/collision detection • Master–slave systems • Multiplex solutions – Time-multiplex – Frequency-multiplex – Code-multiplex • P2P/switched network • Token based	• Clock rates • Modulation and coding • Half/full duplex • Directed communication • Baseband communication	• Transmission media • Differential signaling • EMC immunity • EMC emissions • Signal integrity • Worst-case channel – Impedance – Reflections • Crosstalk • Retransmissions

Note: A fundamental requirement in all cases is that the costs match the advantages of the solution.

driver. This led in 1991 to the I-bus (“Instrumentierungsbus,” English: instrument bus) and in 1993 to the K-bus (“Karosseriebus,” English: body domain bus).

So, BMW used the I/K-bus, Daimler used CAN, Volkswagen used the A-bus [10], PSA and Renault the Vehicle Area Network (VAN, standardized in ISO 11519–2), and the US car makers somewhat later introduced J1850 [6], standardized in ISO 11519–3. Yet another early in-vehicle bus system used in the industry is J1708 [11]. At some point, the car manufacturers realized that the various solutions had more disadvantages than advantages. The volumes were small for the semiconductor vendors and the suppliers had to support different automotive solutions without adding any distinct value to the products.[1]

The next sections will describe the main technologies deployed by BMW today (CAN, LIN, MOST, FlexRay, plus pixel, and consumer links). The descriptions make no claim to be complete, but focus on the aspects the authors consider relevant for understanding in-vehicle networking in the context of Ethernet.

One last general remark: In the end, all networking technologies have to solve the same basic issues. They start with a serial interface that needs to be shared by several users. For these users, the access to the medium needs to be organized, a certain data rate needs to be provided, and the transmission needs to be robust. Table 2.1 gives a rough outline of the topics that need to be addressed and some basic choices that need to be made when designing a communication technology.

2.2.2 Controller Area Network (CAN)

2.2.2.1 Background of CAN

The Controller Area Network (CAN) was one of the first in-vehicle networking technologies to have been developed, but, in contrast to other early in-vehicle networking technologies, it continues to be used. Its development started at BOSCH in 1983 and

Table 2.2 Overview on the CAN ISO standard(s)

Identifier	Content	Year of release
ISO 11898–1	Data link layer and physical signaling, identical to BOSCH CAN 2.0 specification	2003
ISO 11898–2	High-speed medium-access unit up to 1 Mbps	2003
ISO 11898–3	Medium-access unit for low-speed, fault-tolerant Media-Dependent Interface (MDI) up to 125 kbps	2003
ISO 11898–4	Time-triggered communication	2004
ISO 11898–5	High-speed medium-access unit with low-power mode	2007
ISO 11898–6	High-speed medium-access unit with selective wake-up functionality	2013
ISO 16845–1	CAN conformance test plan: Data link layer and physical signaling	2015
ISO 15765–2	Diagnostic communication over CAN: Transport protocol and network layer services	2016

in 1987 the first CAN controller was presented to the public. Daimler was the first car manufacturer to introduce CAN in 1992 [12]. In 1993 the first CAN ISO Standards were published: ISO 11898 on protocol and High-Speed PHY layer and ISO 11519–1 on protocol and Low-Speed PHY layer. Since 2003, the ISO standards have the structure as shown in Table 2.2. In 2016 the CAN bus was the most widely used in-vehicle networking technology, and is also used in other industries like industrial and building automation, aerospace, medical engineering, etc. Nearly every car model is equipped with a CAN bus [10]. This might seem surprising, as only one company owns all the intellectual property and, at least to begin with, did not rely on any standardization organization for its distribution. In the authors' opinion, the reasons for the success of CAN are as follows:

1 **BOSCH decided on an open licensing policy.** The technology was brought to a standards setting organization relatively early and the key elements for the licensing model are easily accessible to anyone interested [13].
2 **Early cooperation with leading semiconductor companies** ensured a good product portfolio for the automotive industry. Intel, NXP (then Philips Semiconductors), and Freescale (then Motorola, now NXP) introduced their first CAN controller products in 1987/88 [12] [14].
3 **BOSCH is a customer for its own technology.** As one of the largest automotive suppliers [15] BOSCH is active in various roles in the automotive sector. The scope ranges from chip manufacturer to holistic system supplier of automotive components. BOSCH thus has a significant impact on the features provided in the chips of their own semiconductor division, as well as in the automotive semiconductor industry as such (see also Section 2.3 for the car manufacturer to supplier relationships). BOSCH also proved to have the respective stamina and strategy in place to make such a technology successful.

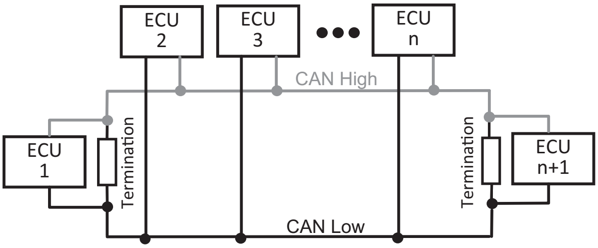

Figure 2.3 Typically used CAN topology ("linear" topology). With the help of passive coupling elements, it is also possible to realize more starlike topologies. The coupling elements function as a sort of ground that separates the different branches. However, today, their deployment in automotive is not very common.

4 **Partners in the ecosystem** engaged in completing the usability of the technology by, e.g., providing test specifications and thus enhanced the acceptance of the technology in the community [16].
5 Last, but not least, **the technology proved to be robust** and usable in all areas of the difficult automotive environment (for details on the latter see also Section 4.1) with one company taking the responsibility for it.

2.2.2.2 CAN Technology

CAN is a bus system, i.e., all Electronic Control Units (ECUs) are attached to and thus share the same wiring (see Figure 2.3). The key element of CAN is its method to decide which unit gets access to the medium/bandwidth. The method CAN uses is referred to as "arbitration" and functions on the principle that the message – and thus the ECU sending that message – with the highest priority/lowest value identifier can transmit. The method is called arbitration, because it is at the moment of transmission that the message with the highest priority wins over competing messages with lower priority.

The idea behind the CAN arbitration is based on pure electric principals. The arbitration distinguishes between "dominant" (0) and "recessive" (1) bits in the message identifiers. The dominant bits, which electrically result in a low ohmic resistance on the channel, override the recessive bits, which result in a high ohmic resistance. So if two ECUs start transmitting simultaneously, the ECU whose message starts with the larger amount of dominant "0" bits succeeds. In other words, the more "0s" a message identifier starts with, the higher its priority. As soon as a unit perceives that the message on the bus is no longer the message it is sending it stops its own transmission, waits for the actual transmission to terminate, then waits for the expiration of the interframe gap and tries to send its message again. This bears the risk that a message with a lower priority never has the chance to succeed if the network is very busy. A good rule of thumb is to design the CAN bus for a maximum load of 50% [4].

Figure 2.4 shows a simplified circuit diagram of a CAN transceiver (here for the High-Speed CAN, HS CAN version). It visualizes the electrical principles behind the arbitration. CAN uses a "Non-Return-to-Zero" coding, meaning that its symbols are

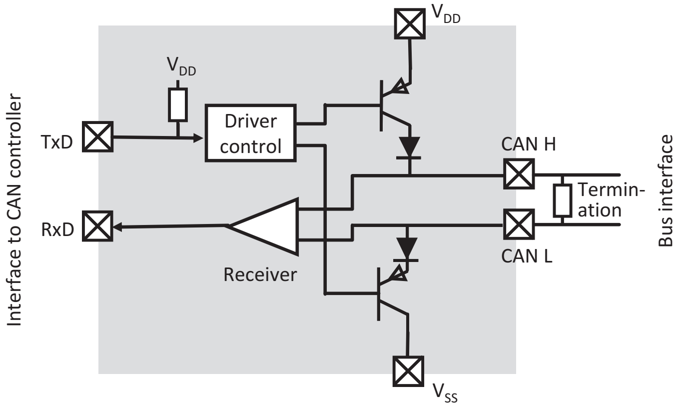

Figure 2.4 Simplified circuit diagram of a HS CAN transceiver.

represented by constant voltage levels on the channel. In the case of CAN there are two levels. In the case of a dominant "0," CAN_H is actively pulled onto VDD and CAN_L is pulled on VSS, which results in somewhat lower voltage level of VDD – VSS. In the case of a recessive "1" the transistors are inactive and the voltage on the bus will stabilize around (VDD – VSS)/2. The key is that the system is not symmetric in its behavior. The change from "1" to "0" is active and immediate. The change from "0" to "1" happens as a discharge in the complete network. This results in a timing behavior that depends on the network, especially the lengths and number of stubs as well as the terminations and their locations in the network. To function properly, the receivers have to be able to perceive the "1" on the channel before the next bit is sent. However, the network is not perfectly synchronized and, as explained, the propagation of a "1" requires time. So, if the network is too large, or the data rate is too high, the transmission becomes erroneous.

Because of this the transmission rate during the arbitration is generally used at 500 kbps and not at the 1 Mbps the standard implies. The dependencies and thus limitations of this network are inherent in the CAN network. The number of ECUs is not per se limited by this. It is their location and termination that affects the timing behavior in the network. The number of ECUs is limited by the driver output.

Because of its focus on messages CAN is sometimes referred to as a "message-based" system. When a vehicle is developed, all possible messages on the CAN bus and their priorities have to be defined upfront. The priorities get encoded into "identifiers" and the developers can choose either a system that supports 2^{11} different identifiers or a system that supports 2^{29}.

Nowadays, there are three different CAN versions: the High-Speed CAN (HS CAN), the Low-Speed CAN (LS CAN), and CAN FD (CAN with Flexible Data rates). The HS CAN can be used for gross data rates up to 1 Mbps, but for the reasons given above is generally used at 500 kbps. The LS CAN can be used for data rates up to 125 kbps.[2] Additionally, BOSCH has recently launched CAN FD (CAN with Flexible Data rates), designed for data rates above 1 Mbps [17].[3] CAN FD achieves the higher data rates

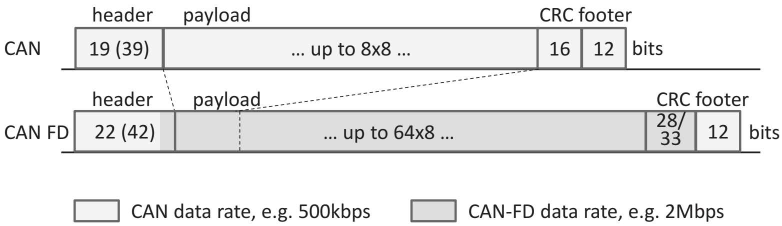

Figure 2.5 CAN FD packet structure in comparison to a HS CAN packet; header length increases by 20 bits in case a 29-bit identifier is used; the CRC length for CAN FD varies depending on the length of the payload.

by allowing for payloads up to 64 bytes – LS and HS CAN payloads can have 8 bytes only – and by transmitting the payload at a higher clock rate. Owing to the physics behind the arbitration as described, the data rate during arbitration remains at HS CAN level. To ensure the robustness of CAN FD, the payload is protected by a more powerful channel coding mechanism than is used for the HS CAN [17]. Figure 2.5 depicts the difference in the packet structure and data rate use between a HS CAN and a CAN FD packet. Table 2.3 shows the elements that make up a CAN packet.

In all CAN versions a differential (also called "symmetrical") signal is transmitted. This suppresses common interferers and thus improves robustness and EMC performance. The arbitration mechanism is also the same for all versions.

CAN does not preclude two different ECUs from using the same identifiers. It is therefore the implementer who has to make sure that every ECU receives its own unique set of identifiers. The payload of a traditional CAN packet consists of 8 bytes, the one of a CAN FD packet can have up to 64 bytes. A transport protocol that enables longer messages spread over several packets has been standardized in ISO 15765–2, which has recently been updated in order to accommodate for CAN FD. CAN does not foresee any addressing. A transmitter puts its message on the bus and all connected ECUs can potentially receive it. It is defined in the receiver whether a message identifier triggers the receiving ECU to store and process the offered data or not.

All participants on the bus acknowledge the reception of every error-free CAN frame received with a dominant bit in the same acknowledge space in the packet, independent of whether they actually use the data. The transmitter in return, will recognize the acknowledgment. The uncertainty in the CAN system is that one acknowledgment is sufficient for the transmitter to perceive a correct transmission. The transmitter cannot discern, which unit(s) have sent the acknowledgment and if the intended receiver unit was among them [19].

Figure 2.6 shows the elements needed for a CAN communication. Generally, the transceiver is a separate semiconductor, while the controller is integrated into the microcontroller. The clock rate, i.e., the transmission rate, is determined by each controller; there is no synchronization in the network other than what can be evaluated from observing the traffic on the channel. An important advantage of CAN is its robustness. It allows ECUs to be connected in almost all areas of a car.[4] For wiring Unshielded Twisted Pair (UTP) cables and multipin connectors can be used.

Table 2.3 Structure of CAN packets (also called "CAN messages") [18]

Arbitration part Field name	Length (bits)		Details
	11 bit id.	29 bit id.	
Start of frame	1	1	Denotes the start of frame transmission (always 0)
Identifier (A)	11	11	First Part of the unique identifier that includes the message priority
Remote Transmission Request (RTR) or Substitute Remote Request (SRR)	1	1	RTR normally 0 ("dominant"), 1 ("recessive") in case of "Remote Frames";[a] 1 ("recessive") if SRR and 28-bit identifier.
Identifier extension bit	1	1	0 ("dominant") for 11-, 1 ("recessive") for 28-bit identifier. In case of 1, next two fields are added.
Identifier B	n/a	18	Second Part of the unique identifier for the data including the message priority.
Remote Transmission Request (RTR)	n/a	1	Normally 0 ("dominant"), 1 in case of "Remote Frames"[a]

Remaining packet Field name	Length (bits)		Details
	CAN	CAN FD	
Flexible Data rate Format (FDF)	n/a	1	Recessive 1 plus next bit dominant 0 initiates the FD
Reserved bits (r0, r1)	1 or 2	1 or 2	Reserved bits (1 in case of 11-, 2 in case of 29-bit identifier)
Bit Rate Switch (BRS)	n/a	1	Recessive 1
Error State Indicator (ESI)	n/a	1	Recessive 1 (if error passive)
Data length code	4	4	Number of bytes of data (0–8/64 bytes)
Data field	8×8	8×64	Data to be transmitted (length as specified before)
CRC	15	27 or 32	Cyclic redundancy check, for CAN FD the 32-bit CRC is used in case the payload $>16 \times 8$ bit
CRC delimiter	1	1	Must be 1 ("recessive")
ACK slot	1	1	Transmitter sends 1 ("recessive") All receivers send a 0 ("dominant") ACK, if they have been able to receive the packet in this same slot
ACK delimiter	1	1	Must be 1 ("recessive")
End of frame	7	7	Must be 1 ("recessive")
Intermission	3	3	

[a] "Remote frames" allow for polling the transmission of data from another unit. They are not commonly used.

2.2.3 Local Interconnect Network (LIN)

2.2.3.1 Background of LIN

There are many applications inside a car which require only simple sensor–actuator communication and the robustness of a 1-wire communication system. The use cases comprise comfort functions like power windows, central locks, electronic mirror

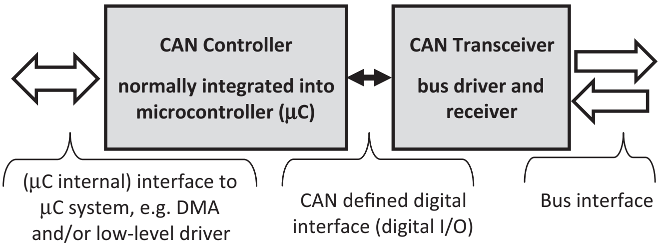

Figure 2.6 Elements and interfaces of a CAN node.

adjustments (see also Section 2.1), electronic seat adjustments, rain sensors, light sensors, electric sunroofs, control of air conditioning, etc. These application areas are very cost sensitive and have few requirements. CAN overperforms for these applications and is thus deemed too expensive.

With the discussions that arose around the first in-vehicle networking systems in the industry, car manufacturers noticed that many had the same requirements for a simple networking technology. Thus, in 1998 Audi, BMW, Daimler, Volkswagen, Volvo, Freescale (originally Motorola), and Mentor Graphics (originally Volcano) founded the Local Interconnect Network (LIN) consortium to standardize a respective solution [20]. In the end, this was the turnaround for the industry, away from local/individual solutions toward commonly deployed standards. The accepted version, LIN v1.3, was published in November 2002, and LIN 2.0 followed in September 2003 [21]. In the US, the SAE published J2602 in September 2005, a LIN 2.0 version with some minor deviations to supposedly even better meet the cost targets [20]. In 2013 LIN was transferred to ISO 17987 [22].

2.2.3.2 LIN Technology

The key requirement of LIN was to be cost efficient, which naturally influenced the technology. One of the first choices was to base the Physical Layer on the K-Line ISO 9141 standard, which had been known in the industry from the early days of in-vehicle diagnostics (see also Section 2.2.1). LIN was designed as a single-ended, i.e., 1-wire unshielded, system with several consequences:

1 The effort to provide LIN hardware was small.
2 LIN is single-ended and not differential/symmetrical. Common noise can affect the system, which limits the immunity and increases the emissions. To nevertheless meet the EMC requirements the data rate is limited to 19.2 kbps. This in return means that a basic clock synchronization mechanism and simple drivers are sufficient.
3 Ground is used as a back channel. This means that the system needs to be designed such that a certain amount of ground shift can be supported. At the receiver the recessive “1” state is thus defined for when a voltage level above 0.6 V_{Batt} has been detected. The dominant “0” state is detected when the voltage level is below 0.4 V_{Batt} [21].

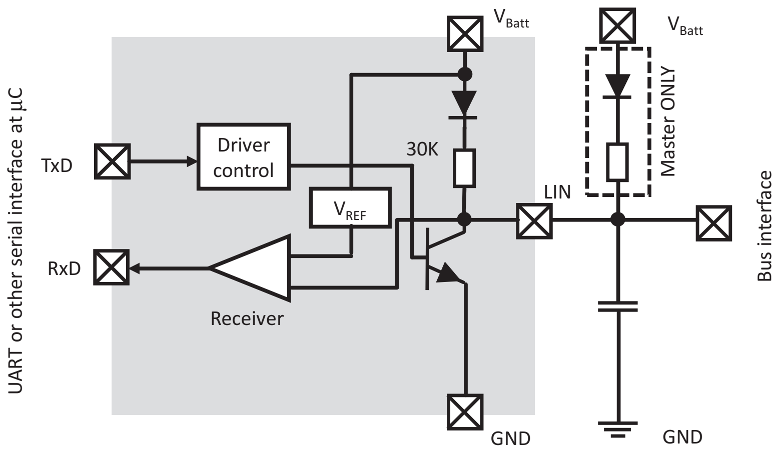

Figure 2.7 Simplified circuit diagram of a LIN transceiver.

4 The behavior of the network also depends on the layout. So the physical expansion of the LIN bus inside the car should be restricted accordingly.

Figure 2.7 shows the circuit diagram of a LIN transceiver. The differentiation between a Master and a Slave node (see below) is achieved by external components. The Master terminates the complete network.

LIN is a good example of how a simple bus system can be derived from enhancing an existing serial interface with a communication protocol. LIN has been designed such that up to 16 ECUs can share the media the bus provides. The multiple user access protocol is a master–slave concept for the channel access, i.e., one unit on the bus is assigned Master (see also Figure 2.8) and Slaves can transmit only after having been polled by the Master with a respective header. As LIN is a bus system, a master initiated communication can also happen between two slaves – all information on the bus can potentially be read by any unit attached to it.

Additionally, the scheduling of the communication on the LIN bus is predetermined in the design phase of a vehicle. The scheduling tables are transmitted to all LIN ECUs in so-called "LIN Description Files." With LIN 2.0 a transport protocol was defined. This means that one message can be transmitted via several packets – each packet has a maximum payload of 8 bytes – and in consequence LIN can also be used for diagnostics

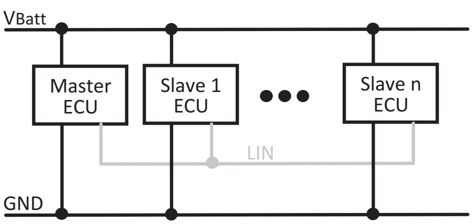

Figure 2.8 Example of a LIN network ($n < 16$).

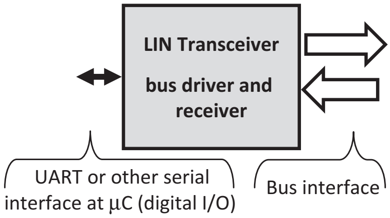

Figure 2.9 Elements and interfaces of a LIN node.

and software updates. It is not uncommon that a more complex ECU is connected to several LIN busses and hosts several LIN masters.

Figure 2.9 shows the elements and interfaces needed for a LIN node. As can be seen, LIN only requires a comparably simple transceiver. To connect to the microcontroller a UART interface is sufficient. A UART interface is so common that it is supported by even the smallest microcontrollers. Slaves can synchronize independently, and do not need an external clock. Furthermore, LIN uses the battery voltage on the channel and thus does not require voltage regulators (see Figures 2.7 and 2.8).

Because LIN achieved its goals of cost efficiency and unifying the industry to this solution, LIN has been successfully established in the industry. The use cases that it addresses will not disappear anytime soon from vehicles and LIN can thus be expected to persevere in the in-vehicle networking landscape. LIN is an example that proves that there is likely never going to be one ideal in-vehicle networking solution, even if Automotive Ethernet can address many application areas and might reduce the number of technologies prevailing.

2.2.4 Media Oriented Systems Transport (MOST)

2.2.4.1 Background of MOST

At the end of the 1990s the need to support complex audio applications in cars became urgent. Not only were audio CDs irreversibly replacing analog music storage media, but the customers were also getting more interested in navigation systems and mobile phones usage. Inside the car the different audio streams these applications caused had to be coordinated with each other and with additional warning notices from driver assist functions, while at the same time providing an optimum sound experience.

This first of all required a significant increase in the data transmission rate and it became obvious relatively quickly that only an optical system would solve the task. At that time optical systems were the only systems promising the expected data rate with an EMC compliant solution at a reasonable cost level. Daimler already had experience from using the Domestic Digital Bus (D^2B),[5] a technology later standardized as IEC 61030. The Daimler experience was particularly valuable as it proved the feasibility of the less costly and more robust Polymeric Optical Fiber (POF) as a transmission media in contrast to Glass Optical Fiber (GOF).

Nevertheless, the intention was not only to have a new PHY technology with a higher data rate available, but to have a system that covered all networking aspects, from the physical layer to the application layer. The technology was to enable the complex control sequences the desired use cases needed in the already challenging automotive environment. This was unprecedented in the industry, and was seen as a challenge that required a new approach.

The industry decided to cooperate with a strong supplier, who in the role of a type of "general contractor" was to coordinate the fortunes of the new networking system, while the requirements were still provided by the car manufacturers. The partner chosen was Oasis (which later became part of SMSC, which today is part of Microchip). Some of the founding members of Oasis had previously worked for (Harman) Becker, and thus had significant experience with automotive infotainment systems and their requirements. Additionally, Oasis had a proposal for a technology supporting the optical transmission. So in 1998, BMW, (Harman) Becker, Daimler, and Oasis founded the MOST Cooperation (MOST Co) to develop MOST as a networking technology for in-vehicle applications and to establish MOST as an industry standard. Other interested companies were welcome to join the MOST Co. The agreements include royalty-free licensing among the members for all developments except for the Data Link Layer (DLL)/PHY, for which the IPR belonged to Oasis and thus today to Microchip. To license the DLL/PHY technology a royalty-based license was/is possible but also necessary.

The MOST technology is often associated with a monopoly, as indeed there is only one supplier of the respective hardware (Microchip). With the selection of a supplier in the role of a "general contractor" the participants never intended to have a closed market and the MOST Co was set up in a way that allowed for competition. This was not the focus, however. The main goal was to minimize the risks associated with the development of such a complex system, and having a "general contractor" seemed to minimize those risks. One of the reasons given today for the missing competition is that the MOST Co did not provide a specification to which interoperability and compliance testing of the DLL could have been performed, but that Microchip would have licensed an IP core. Potential other vendors thus had limited chances to differentiate their product and face(d) the risk of incompatibility to the technology of the dominant vendor, Microchip, if they tried. This had been a problem in the past, e.g., when Token Ring failed, because it was dominated by IBM and second source vendors were never sure that they had a chance to be compatible on all levels in order to make their systems work [23]. The automotive market is somewhat different to the computer industry. Nevertheless, because the MOST Co focused on doability, the important issue of interoperability was missed at the time and the fact is that no other company took the opportunity to enter the market.[6] At the time of writing, the ISO project 21806 was being started in order to transfer the MOST specifications, including the DLL specification, to ISO.

MOST exists in four variants: MOST25 with optical transmission first introduced by BMW in 2001; MOST50 with electrical UTP cabling first introduced by Toyota in 2007 [24]; MOST150 first introduced in the optical version by Audi in 2012 [25]; and MOST150 using coaxial cabling, which is yet awaiting market introduction. As the

Table 2.4 Structure of a MOST control message

Element	Length (bytes)	Content
Priority decision	4	The value of this field allows a unit to identify whether it can send a control message. Priority of the message, availability of the media, and transmission history are taken into account
Target address	2	Address of the ECU, which requested the function. As the data is available on the bus, nevertheless all units can listen to it
Source address (DeviceID)	2	Address of the ECU, which offers the function. The address is defined in the network layer and depends on the position of the unit in the network
Message type	1	Identifies the type of control message
FBlockID	1	ID of the function block
InstID	1	Instance inside the FBlock. One FBlock can consist of several instances
FktID	1½	Function to be called
OpType	½	Defines the function type, e.g., could be an error message or a request
Tel ID	½	Identification of parameters
Tel Len	½	Length of parameters
Data	12	Content of the parameters
CRC	2	
ACK	2	Defines the function type, e.g., could be an error message or a request
Reserved	2	
Overall	32	

Note: From “FBlockID” to “Data” represents the data field of the control message.

following discussion is concerned with the basic principles only, it will use MOST25 as an example.

2.2.4.2 MOST Technology

The MOST technology defines the communication on all seven layers of the ISO/OSI layering model. The MOST protocol is thus more complex than for the previously described CAN or LIN. MOST is the first in-vehicle networking technology to support service-based methodologies, which means that functions and services can be requested during operation on demand. The interfaces to the available functions are described in detail in the MOST Function Blocks (FBlocks). Also, for the first time, message sequences are provided. On the higher layers a MOST message is defined to consist of the following parts: DeviceID.FBlockID.InstID.FktId.OpType.Length. Table 2.4 lists how the elements of a control message form a 32 byte message.

A MOST message is not identical with a MOST frame. Instead a message might have to be distributed over several frames. Frames represent the constantly repeated structure in which the traffic on the MOST bus is organized, with the bus topology generally

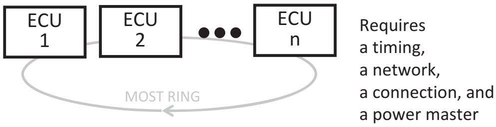

Figure 2.10 Example MOST ring ($n < 65$).

being a (physical or virtual) ring. One frame is partitioned in 64 slots, which equals 64 bytes. Two of these are administrative and two are for control information. This means that the 32 byte control message of one unit is spread over 16 frames and that the next unit has to wait before it can use the 2 bytes "control channel." MOST supports the two system frequencies: 48 kHz, from professional audio and 44.1 kHz, from audio CDs [26]. In case the frames are sent at 44.1 kHz the bandwidth for MOST25 that can be shared on the control channel is 705.6 kbps. The reception of control messages needs to be acknowledged.

The remaining 60 bytes, i.e., 21.2 Mbps@44.1 kHz or 23 Mbps@48 kHz, can be divided into a synchronous and an asynchronous part. Note that once a MOST25 system has been set up, the division between the two parts is not changeable during operation.

- **Synchronous data:** MOST has been optimized to transmit audio in continuous data streams; hence the frame frequencies of 48 kHz for general audio or 44.1 kHz for audio CDs. 24 to all 60 of the available bytes can be attributed to synchronous data. The multiple access for these bytes is organized in Time Division Multiplex, i.e., in every frame a certain unit transmits its data in certain byte(s)/slots. In MOST jargon the assigned slot in the synchronous section is called a "channel." There are no retransmits for lost packets.
- **Asynchronous data:** 0 to 36 bytes can be used for the transmission of application data like map information from navigation systems or TCP/IP traffic. This corresponds to a maximum gross data rate of 12.7 Mbps@44.1 kHz and 13.8 Mbps@48 kHz. Access to the channel is granted by a token system. When a unit has the token, it can use all available bytes assigned to asynchronous data. A unit can transmit one message with a maximum of 1014 bytes user data, before it passes the token on. As for control messages, such a message is transmitted over several frames – 29 frames, if the unit received the theoretical maximum bandwidth. There are no acknowledgments or retransmits.

MOST generally uses a ring topology, virtual or actual (see Figure 2.10 for an example), which can handle up to 64 ECUs. Each ECU is addressed according to its location in the ring. One ECU functions as a "timing master" that continuously sends the preamble that begins every frame onto the ring and that allows all ECUs to synchronize. The additional coordination functions of network, connection, and power master do not have to be handled by the same unit but generally are. Connection master administers the synchronous channels, the network master controls the system status, and the power master supervises start-up and shutdown of the MOST network.

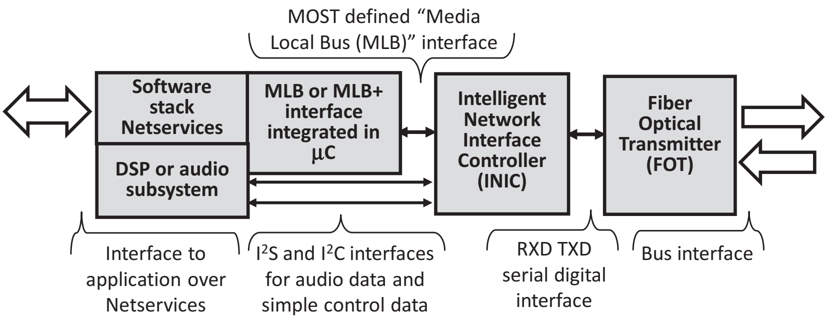

Figure 2.11 Elements and interfaces of a MOST node.

Figure 2.11 shows the elements needed for a MOST communication node. As can be seen, the capabilities, but also the complexity have increased significantly. The communication functionality is provided by MOST network services, for which now Microchip owns the trademark "Netservices." The PHY is represented by the Fiber-Optic Transmitter (FOT), the DLL by the MOST Network Interface Controller (NIC). The Intelligent Network Interface Controller (INIC) is controlled by the Netservices, which are associated with the network, transport, and session layer of the ISO/OSI layering model [27]. The application sockets and FBlocks share layers 6 and 7. With the boundaries between the layers blurred it is not practical to exchange protocols or even adjust the functionality on individual layers but to use the complete stack as is.

For efficient use of the INIC it is advisable to use the Media Local Bus (MLB). Since MLB was defined with MOST, it is not a very common interface at host processors in an ECU. Thus, in some cases the use of an additional companion chip might be required (see Figure 2.11). The audio data (and simple control functions) can use the Inter-IC Sound (I^2S) and Inter-Integrated Circuit (I^2C) interfaces. The FOT converts the electrical information into an optical signal. For MOST the communication on the fiber is unidirectional only.

2.2.5 FlexRay

2.2.5.1 Background of FlexRay

FlexRay was developed at a time when the automotive industry became interested in "X-by-Wire" applications. The idea of X-by-Wire is to eliminate all mechanical fallback from the car and to have pure electric functions only.[7] Target applications were the steering, braking, and other safety critical systems. Security and timing are particularly important in this case. BMW had already gained some experiences with time-triggered communication prior to the development of FlexRay with the proprietary development of a technology called Byteflight. This optical system was used in a few BMW models for airbag control and other safety related systems. Nevertheless, it did not persevere as it proved too expensive.

Instead, in 2000 BMW and several other automotive companies agreed on developing a new technology in the FlexRay Consortium. The core partners were Freescale

Table 2.5 Overview on FlexRay standards

Identification	Content	Year
FlexRay Protocol V3.0.1	Data Link Layer Specification, designed to be fully backward compatible. Most car manufacturers use Version V2.1 Rev A	2010 (V2.1, 2005)
FlexRay Electrical Physical Layer V3.0.1	Physical Layer Specification, designed to be fully backward compatible. Most car OEMs use Version V2.1 Rev B	2010 (V2.1, 2005)
ISO-17458–1	General information and use case definition	2013
ISO-17458–2	Data Link Layer specification	2013
ISO-17458–3	Data Link Layer conformance test specification	2013
ISO-17458–4	Electrical Physical Layer specification	2013
ISO-17458–5	Electrical Physical Layer conformance test specification	2013

(formerly Motorola, now NXP), NXP (formerly Philips), BMW, Daimler, and somewhat later, BOSCH, General Motors (Opel), and Volkswagen [28]. In 2009, after the finalization of FlexRay 3.0, the task was seen as completed. The FlexRay Consortium was disbanded [29] and the FlexRay standards were transferred to ISO (see Table 2.5 for an overview).

2.2.5.2 FlexRay Technology

As can be seen in Table 2.5, FlexRay defines the Physical (PHY) and Data Link Layers (DLL) only. Other layers are covered by other committees, for example the AUTomotive Open System ARchitecture (AUTOSAR) standardization explicitly addresses the higher layer software protocols needed for a FlexRay communication. The key requirement for FlexRay is reliability and consequently FlexRay provides a number of respective features like determinism and redundancy.

First, FlexRay communication is based on timeslots and cycles, which are configurable by the developer. Every cycle consists of a static time segment and a network idle time, but can additionally comprise a dynamic time segment. The multiple user access is handled differently in the static and dynamic segments: In the static section, the access is defined in TDM, i.e., the units get assigned certain timeslots in every cycle up front. If a unit has nothing to transmit in its timeslot in the static segment, it transmits a “null” frame, so that the respective receiver always receives something as expected and knows that the communication is not unintentionally disrupted. The dynamic segment uses a so-called “mini-slot” method, which uses a preset order of FrameIDs combined with counters for multiuser access. Other than in the static segment, however, a unit whose turn it is to transmit does not do so unless it has data to send. In this case, all units increase their counters and the one having the next FrameID can start the transmission in the next mini-slot instead of having to wait. To increase the throughput the mini-slot duration is shorter than the static slot and can be configured with the design of the system [28]. The mini-slot method is a heritage from the Byteflight development.

Table 2.6 shows the setup of a FlexRay packet. Each packet consists of a header, a payload, and a trailer, which comprises the CRC for the payload. The gross data rate is 10 Mbps. How much effective data rate a system has, depends on the configuration

Table 2.6 Elements of a FlexRay packet [28]

Field name	Length (bits)	Detail
Reserved bit	1	Reserved bit
Payload preamble indicator	1	Information whether data packet with payload
Null frame indicator	1	Information whether data packet is a Null frame
Sync frame indicator	1	Information whether data packet is a Sync frame
Start-up frame indicator	1	Information whether data packet is a Start-up frame
Frame ID	11	Packet identifier
Payload length	7	Amount of data to be transmitted
Header CRC	11	Covers Null frame indicator to Payload length
Cycle count	6	Indicates the actual cycle
Data field	0–254 bytes	Transmitter data is signaled by "1" (recessive). A receiver NACK is signaled by "0" (dominant)
Payload CRC	24	CRC for payload
Overall	8+(0–254) bytes	

of the system and the ratio between the lengths of the dynamic and static sections. Additionally, the used 8B10B Non-Return-to-Zero (NRZ) coding as well as header and trailer reduce the net bit rate [30].

Like CAN, FlexRay also transmits a differential signal. In contrast to CAN though, the FlexRay transceiver has two separate push/pull entities (see also Figure 2.12). This means that as well for the high as for the low voltage level the current is actively driven and that the behavior of the signaling is less influenced by the layout of the network for FlexRay than it is the case for CAN. Nevertheless, FlexRay provides a tenfold bit rate (or 20-fold when considering that CAN is normally used at 500 kbps), so higher investment into the network infrastructure and the terminations is necessary in order to meet the EMC requirements. This is especially so as FlexRay uses, like LIN and CAN,

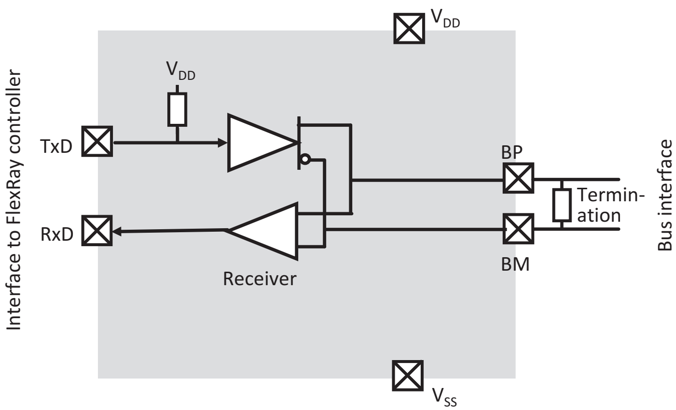

Figure 2.12 Simplified circuit diagram of a FlexRay transceiver. The transmitter circuit has not been included for complexity reasons.

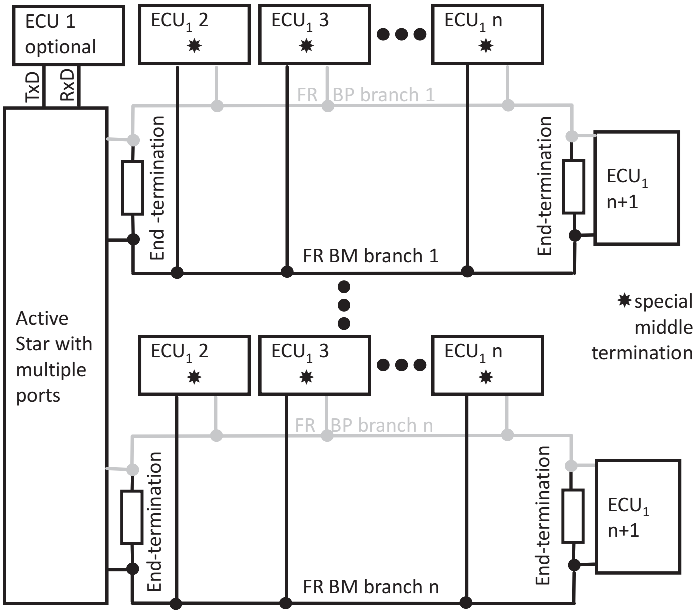

Figure 2.13 Large FlexRay network with active star.

unshielded cabling, and multipin connectors. The cables need to be of better quality and fewer ECUs can be connected to one branch, as will be explained in the following.

A small FlexRay system consists of four or five ECUs using a linear topology. Nevertheless, FlexRay also offers the possibility for different, larger topologies. With use of an "active star/star coupler" several linear topologies can be combined to one network (see Figure 2.13), though it is not possible to cascade the architecture. The star coupler refreshes the data into the other lines without adding noticeable latency. The star coupler also ensures that only one unit in one branch transmits at the time, while in all others the units are in listening mode. What is special is that the star coupler does not use scheduling but observes the voltage levels on the channel. The star coupler is thus also called "moderator." The challenge is the speed of signal propagation, which can result in collisions that the star coupler cannot resolve. The star coupler is thus a challenging element in a FlexRay network.

The elements needed to set up a FlexRay node reflect the lower two layers of the ISO/OSI model FlexRay specifies. The PHY is represented in the FlexRay transceiver. The communication logic is handled by the communication controller (CC), see Figure 2.14. The CC also handles the timing synchronization and regulates the clocks onto the "FlexRay" time. Normally, the CC is integrated into the microcontroller.

FlexRay is an in-vehicle networking technology well suited for power train and chassis control. Nevertheless, its use did not quite develop as expected [31]. Safety critical "X-by-Wire" applications are evolving very slowly; by 2013 only Nissan had publicly announced the introduction of an "electronics only" steering [32]. Also, FlexRay did not really prove suitable as an in-vehicle backbone, because the tightly synchronized

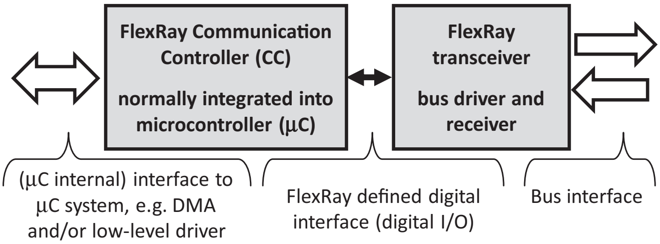

Figure 2.14 Elements and interfaces of a FlexRay node.

packets are challenging to handle in the software of the ECUs. This could be eased by using a synchronized Operating System (OS) like OSEK Time. Nevertheless, at the time of writing OSEK Time was not common in automotive. Whether the possibility to set up FlexRay with redundancy, i.e., a second link, is going to be exploited in the industry is unclear.

2.2.6 Pixel Links

The motivation to develop MOST25 was to be able to handle sophisticated digital audio applications. Compared with the available in-vehicle networking technologies at the time of development, MOST25 provided a very large increase in data rate. Nevertheless, the data rate of even an uncompressed audio stream is small when compared with high-definition (HD) camera, video or graphic display data.

The data rate of such HD data is derived by four aspects: The pixel resolution of the camera imager or display, the bit depth which encodes the colors, and the rate at which the image frame is renewed (frames per second, fps). A very high data rate currently being discussed in the context of HD videos stored on a Blu-ray disk uses a resolution of 3840 × 2160 pixels [33]. With a color depth of 20 bits and 60 this results in a data rate close to 10 Gbps. More common for HD are pixel resolutions of 1280 × 720 or 1920 × 1080 [34] [35], which depending on the color depth and frame rate results in data rates between 0,22 Gbps and ~3 Gbps (see also Section 4.3.3.1 for more details).

Whether you actually need to transmit any of these data rates in the network, depends on the Electric and Electronics (EE) architecture. Figure 2.15 shows example use cases, in which data rates above 1 Gbps occur. Recorded video, camera data, or a graphics processor might be the source. Figure 2.15 deliberately does not show which of these three blocks for each use case are in the same ECU, i.e., which units are connected on a circuit board and which require the transmission link. For the example of Blu-ray (Figure 2.15(b)) the disk is read and the data is transmitted at 54 Mbps only [36]. The decoder needs to be integrated with the display, which is likely to be at a different location. However, if not a Blu-ray disk is being read but a less protected video format is stored in a device,[8] it is also possible to directly decode the data in the same device and then to forward uncompressed data to the display.

In the case depicted in Figure 2.15(c) the processor is likely to be directly integrated with the camera. The camera can perform image processing and encoding and the data

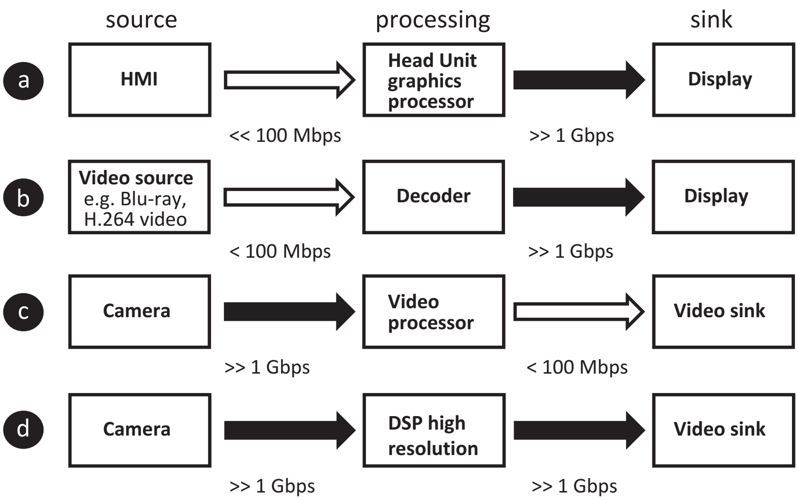

Figure 2.15 Example use cases for high-speed link applications.

rate transmitted to the video sink is well below 100 Mbps. In this case the video sink would need to be able to decode, but does not have to perform the processing. Or the camera sends unprocessed data to a unit that then performs the processing (Figure 2.15(d)) before internally or externally passing unprocessed data on to the display. Often there is a choice and whatever is being selected varies on a number of parameters that include quality concerns (compression losses), technical feasibility, costs, and personal preferences. The conclusion nevertheless is, that there are use cases in which (video) data at data rates significantly higher than 1 Gbps need to be transmitted (see also Section 4.3.3.1).

Such communication technologies that need to transmit high data rates for video data are not only a topic for automotive, but are typical in the consumer industry, too, where most of them originate. In the consumer industry the shift from analog video transmissions like "RGB" or "FBAS" to high-resolution digital video resulted in a variety of different display link standards with more expensive cables, generally for relatively short distances. The distinction made in this book between the connectivity of this section and "consumer links" discussed in Section 2.2.7 is that the consumer links include clearly defined cables, connectors, interoperability tests, and often some higher layer protocols. This book thus refers to "pixel links" for technologies supporting the high-speed transmission of binary data in order to transmit pixel precise information on the lowest layers of the ISO/OSI layering model only.

In first car manufacturer to introduce such a pixel link into series production was BMW. The 2001 BMW 7-series had a central information display. Analog video was not sufficient to provide the respective quality and existing digital in-vehicle networking systems did not support the data rate needed for the expected resolution. The decision was to use a Low-Voltage Differential Signaling (LVDS) link, first introduced into the consumer world in 1994 [37]. LVDS describes the physical principle with which digital

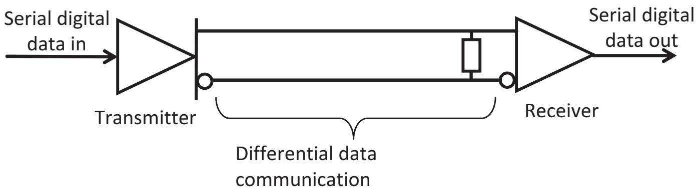

Figure 2.16 LVDS principle.

data is transmitted as a differential signal over a serial link (see also Figure 2.16). To support the high data rates, correct termination is important, and shielded cables are being used. Additionally, de-emphasis are used in the transmitter [38].

LVDS is a physical principle that defines the voltage levels, but it is not a standard. The actual realization with data rates, transmission gaps etc. varies from vendor to vendor, so that various noninteroperable solutions exist on the market. Furthermore, there are plenty of enhancements of the principle technology available. Today, pixel links are, e.g., current driven. This means that the information is not reflected in the voltage but in the current level. These systems are based on Current Mode Logic (CML) [38], which means that to continue to refer to them as LVDS is actually no longer correct.[9] One of the newer developments for CML based pixel links is the use of coax instead of shielded cabling [39] and to be able to transmit power over the same coax link.

Furthermore, early pixel links transmitted pixels only. Control data had to use an additional communication technology. In consequence pixel link products started to appear that include a control channel, a backward channel [40], I^2S or I^2C, in order to integrate audio data onto the video link (e.g., [39]), or an Ethernet channel (e.g., [41], see also Section 4.2). Furthermore, products were and still are being differentiated by optimizing them for the different use cases (see also Figure 2.17). Depending on whether a camera or a display use case is being supported, not only different type of data needs to be transmitted aside from the video in different direction, but also the interfaces the products support vary.

The implementer can therefore choose from a variety of pixel link solution to optimize the implementation – which is desirable. However, they are all noninteroperable – which is not desirable. A small solace is that pixel links do not represent a networking

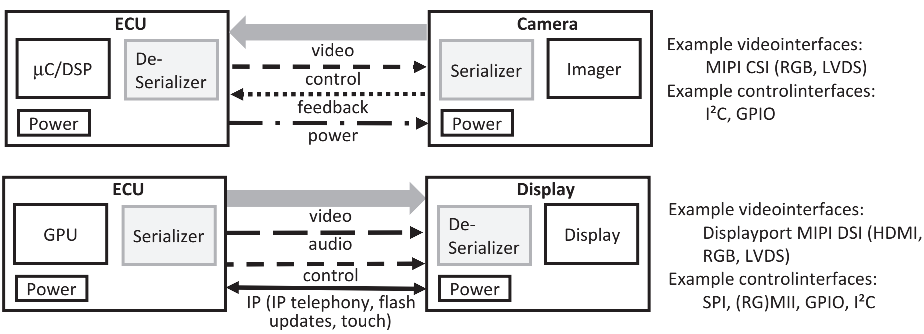

Figure 2.17 Differentiation potential for serializer and deserializer used for pixel links.

technology used for distributing data throughout the car. Pixel links are isolated and not so common connections for mainly unidirectional point-to-point communication on PHY level between two units. In case of supplier change, not the whole network but “only” two ECUs need to be changed.

2.2.7 Consumer Links

The consumer industry is constantly developing new communication technologies. The question often arises: Why not simply use those, especially if the consumer brings them into the car anyway? The answer is that car manufacturers adopt consumer links only where they have to. The reasons are the following:

- **Timeline:** It is not unusual to come across cars that are more than ten years old. For example, in 2015 in Germany about 38% of all registered cars exceeded this age; a percentage that has continuously increased since it has been recorded [42]. Not long ago a car owner simply bought a new car radio to have up-to-date technology inside the car. With the current rate of change in the consumer industry, it is hard to imagine what interfaces a car radio might have to support in ten years; if such thing as a car radio still exists. Therefore, to use consumer technologies for in-vehicle networking would mean working on very unstable ground that the car industry has little control over.
- **Quality:** The quality requirements of the consumer industry are not nearly as stringent as those inside the car (see also Section 4.5.1). If technology of a suitable automotive quality can only be met with expensive cabling and expensive qualification programs, its attractiveness decreases drastically. Obviously, car manufacturers rely on consumer interfaces when integrating consumer devices. This is one of the situations in which the use of a consumer link cannot be avoided. It leads to yet another quality issue: Generally, the perceived quality of the integrated functionalities is associated with the car despite its dependency on the Consumer Electronics (CE) device.
- **Networking functionalities:** A very popular consumer technology that most cars support in one way or other is the Universal Serial Bus (USB). USB is widely deployed and supported in many infotainment and communication related microcontrollers (μCs) and Digital Signal Processors (DSPs). Its use offers itself to the designers and since 2006 USB can be bought as an in-built interface to consumer devices inside the car [43] [44]. Additionally, it is sometimes used for ECU internal communication. Nevertheless, USB was designed to connect peripherals to a computer [45]. The topology it supports, and the communication schemes and networking functions are specific and would require significant costs or workarounds if used for an extensive network inside a car. For example, USB is intended for a star topology with one master, the computer, controlling individually connected slaves, the peripherals. Such EE architecture would lead to extensive wiring inside a car. Additionally, also automotive USB requires the use of expensive shielded cables (see also Section 3.1.2.2) with limited reach. It is a good example of a popular consumer link that is not really

suitable for in-vehicle networking use, though, of course, not all communication use cases in the car require networking function.

- **New requirements:** The digitization of audio and especially video not only results in an increase in quality, but also in an increase in complexity and in at least one function, which is not user friendly: Digital Rights Management (DRM) . DRM requires data encryption and with that causes effort and costs in the components. The use of DRM is neither a choice of the consumer nor of the car manufacturers. Even if a car represents effectively a closed system, car manufacturers are required to provide DRM in their infotainment systems for the respective links, particularly if the communication technology used is one of those easily accessible to every consumer. One copyright protection issue the automotive industry has to consider in this context is High-bandwidth Digital Content Protection (HDCP), which is needed for the High-Definition Multimedia Interface (HDMI) or its evolution, Mobile High-definition Link (MHL).

For the above reasons, car manufacturers are careful when considering the use of consumer links inside the car. For the integration of consumer devices they often have to be supported, but in a clearly defined, limited, and isolated environment. So far none of the wired consumer links has proved to be suitable as an in-vehicle networking technology.[10]

2.2.8 Trends and Consequences

The previous sections described important communication technologies prevailing in vehicles today. The first important message is that each of the technologies described was developed and/or is used with a specific application field in mind: CAN for robust ECU communication, LIN for low cost, MOST for high-end audio, FlexRay for X-by-Wire, pixel links for unprocessed video, and consumer links for consumer device integration. The car manufacturers actively drove some of the standardization work behind these technologies. Table 2.7 shows that this has led to very different technologies, not only in respect to the data rates supported, but also in respect to the communication mechanisms and robustness methods used for the technologies.

Each new use case thus led to new requirements, new standardization efforts, new communication principles, and new qualification processes. This is highly resource binding in development and testing, especially as the technological complexities have increased significantly. Each new technology requires training and specialists who can solve inconsistencies and problems with the technology. Thus, the second important message is: While powerful in-vehicle networking is a fundamental requirement for functional innovations, the number of networking technologies needs to be as small as possible. After all it is the customer experience a customer buys with a car, not the in-vehicle networking technology enabling it. In respect to many of the early developments in in-vehicle networking (e.g., VAN, I/K-bus, K-Line, J1708, Byteflight, and D^2B – see the previous sections) a certain consolidation is already noticeable. In the authors' opinion, Ethernet/IP, while right now just another technology to be introduced into

Table 2.7 Comparison of discussed in-vehicle networking technologies, first overview, outlining the main, high-level differences between Ethernet and the existing (in-vehicle) networking technologies

Technology	Multiple Access Scheme	Data rate	Robustness	Target use case
CAN (FD)	Priority-based messages	Generally 500 kbps (2 Mbps) shared	Differential signal, comparably small data rate	Robust ECU control
LIN	Master–Slave and schedule tables	19.2 kbps shared	Small data rate	Low-cost control
MOST	Priority based, TDMA, token	<25, 50, 150 Mbps shared	Optical for MOST25/150	Complex, high-end audio
FlexRay	(Flexible) TDMA	≪10 Mbps shared	Differential signal	Real-time control, X-by-Wire
Pixel links	None, communication between two partners only	Up to 3 Gbps, unidirectional	Differential signal, shielded cables, short links	Links for unprocessed video
Consumer	None, communication between two partners only	Up to 5 Gbps	Shielded cables, short links	Integration of consumer devices
Automotive Ethernet	Switched, for each link on the network, queuing	100/1000 Mbps per link and direction	Differential signal, intelligent modulation, and filtering	High data rates, use case-independent packets

automotive, has the chance to drive this desired consolidation further,[11] even if it is highly unlikely that there will only ever be one in-vehicle networking technology.

In addition, car manufacturers face the challenge of ever faster changing customer expectations and product diversification in the form of new models, new derivatives, and potentially shorter life cycles. Modularization is a prominent way to handle this [46] [47]. In-vehicle networking technologies have to be flexible and have to support this. If with a derivative a higher speed grade is necessary for a communication link and if with this higher speed grade a completely different technology needs to be used, this is counterproductive. Message three thus is: Automotive requires a future-proof in-vehicle networking technology that is flexible and scalable and can grow with the requirements. More details in this context and Ethernet for in-vehicle use are described in Chapter 3.

One more aspect to consider is that independent from what happens in the automotive industry, communication in general is irrevocably changing. There will always be specific physical environments that need to be addressed, like short- or long-distance links, wired or wireless communication, EMC-sensitive or -insensitive environments, which require special treatment. No matter what the PHY looks like, the application data changes to one type of data: packets. Audio and video is compressed into packets. Circuit-switched telephone networks are being changed into packet-switched networks (see also Section 1.2.4). Internet is packets anyway. As shown in Table 2.7, Ethernet is conceptually different from the traditional in-vehicle networking technologies: It is innately packet based, high speed, and switched.

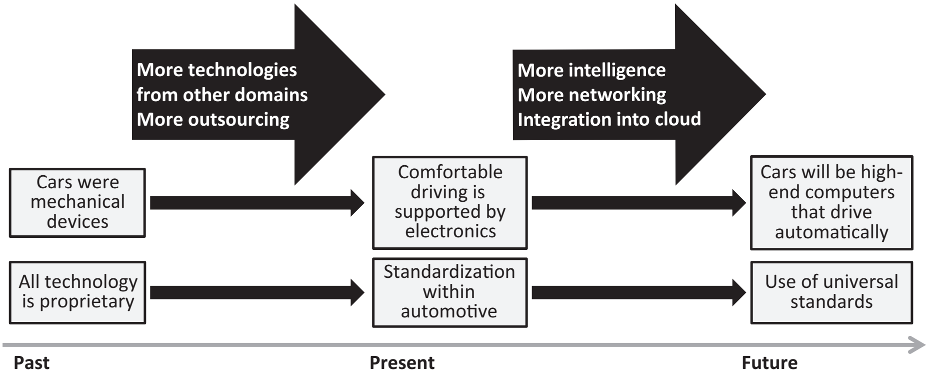

Figure 2.18 Long-term trends in automotive (networking).

Figure 2.18 summarizes the phases the automotive industry is going through and indicates the direction that it is heading. When automotive started, cars were purely mechanical devices. Over time electronics were added and are increasingly replacing traditionally mechanical functions as cars move toward being high-end computers that drive autonomously. At the beginning all the technologies in cars were proprietary. Then the automotive industry started to standardize especially nondifferentiating functions like in-vehicle networking technologies. Nevertheless, the industry is in a constant search for larger economies of scale [48]. With all the electronics, cars are facing the same challenges that have been solved in other industries. The industry therefore moves toward the use of industry-independent standards.

2.3 Responsibilities in In-Vehicle Networking

While the previous sections described the need for in-vehicle networking technologies as well as the technologies themselves, Sections 2.3.1 and 2.3.2 discuss the responsibilities for these technologies in the industry.

2.3.1 Role of the Relationship between Car Manufacturer and Suppliers

Historically, car manufacturers were highly vertically integrated, meaning they produced a great share of components themselves. This is the reason why "Original Equipment Manufacturer" (OEM) is often synonymously used for "car manufacturer." Several developments changed this: The increase of functionalities not inherently related to the driving function; the increased worldwide competition especially from Japanese manufacturers, whose more fuel efficient cars became very attractive with the oil crisis [48]; and the increase of electronics as well as software. As a consequence, vehicle manufacturers started to externalize those components that suppliers were able to deliver at a better value. By today the suppliers represent a well-established part of the automotive value chain and they are responsible for many of the innovations happening in

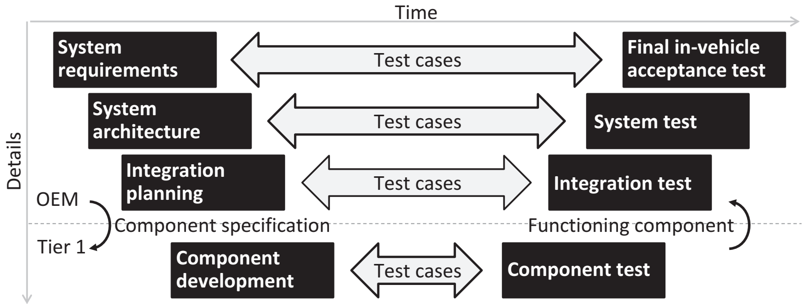

Figure 2.19 Division of responsibilities between car manufacturer and Tier 1 supplier along the V-cycle.

automotive. The car manufacturers generally retain those parts in their own development they have identified as their Unique Selling Point (USP).

Key to the unique customer experience is the composition, design, and overall functionality of the cars. To achieve this with the large number of parts from a large number of sources, these parts need to be precisely defined and assiduously integrated. A single day of delay in the Start of Production (SOP) of a car causes huge losses for the manufacturer. Every single component and its interaction with the rest of the car need to be faultless. Reliability is extremely important.

The V-cycle[12] (see also Figure 2.19) helps to structure the respective division of responsibilities between car manufacturer and Tier 1s for electronic control units (ECUs). The car manufacturer defines the overall system requirements, distributes the individual functions needed to fulfill the system requirements to specific ECUs (i.e., defines the EE architecture of the system, see also Section 6.1), and then defines the interfaces to the rest of the car into which it will be integrated (i.e., the in-vehicle network). The Tier 1 supplier receives the respective component specification. The supplier is then responsible only for the ECU and its functionality according to the specification. The car manufacturer has to do the integration work up to the proof of functionality inside the car.

In the OEM–Tier 1 relationship the component definition is an interactive process. Car manufacturers often want what suppliers can provide (at a reasonable price point) and suppliers often build up that know-how they expect to be reusing with various car manufacturers. This can lead to a chicken and egg problem, if both expect the other to propose new technologies and innovations. This can be solved by a very close development partnership between car manufacturer and Tier 1.

The V-cycle shows two things:

1 **The car manufacturer has a direct business relationship with Tier 1 suppliers only** and not, e.g., with a semiconductor vendor. The semiconductor vendor, who is a Tier 2 has, by definition, only dealings with the Tier 1s. The Tier 1 is the customer of the Tier 2 and the primary source for product requirements for the Tier 2, not the car manufacturer.

2 **It is the car manufacturer who is responsible for the in-vehicle communication.** This comprises the correct design of the distributed functionalities, i.e., a precise definition of the communication interfaces[13] and the provision of the needed data transmission functionalities, i.e., the right choice of the in-vehicle networking technology. This explains why it is the car manufacturers who drive the development of in-vehicle networking technologies (see also Section 2.2 and Chapter 3). The Tier 1 needs to be able to handle the networking technology the OEM requires (meaning they generally have to be able to support more networking technology than every individual OEM [49]) but in the end it is the car manufacturer who is responsible for its choice. It requires Tier 2 semiconductor vendors to provide respective semiconductors, i.e., the basis for a new networking technology. This means that the Tier 2's customer and the responsible decision maker are not the same entity.

Semiconductor vendors do not only play a role in providing semiconductors for networking technologies. In general, many different types of semiconductors enable many innovations in automotive electronics. It is said that 90% of automotive innovations are driven by electronics and software [47] and that the value of semiconductors per vehicle is expected to increase from US$ 250–300 in 2011 to US$ 400–450 in 2020, not counting the value of semiconductors needed additionally in electrically powered vehicles [50]. Thus, not only the Tier 1s but also the Tier 2s play an important role for the innovations in automotive. Like the OEM–Tier 1 relationship the OEM–Tier 2 relationship can be a chicken and egg problem, i.e., the car manufacturer asks the Tier 2 what functions its semiconductors will enable and the Tier 2 asks the car manufacturers what functions they want to have enabled, or it can also be a partnership in which innovations are driven together. What is different in the OEM–Tier 2 relationship is that there is no direct business relationship between car manufacturer and semiconductor vendor. Furthermore, OEMs as well as Tier 1s would like to avoid a monopoly situation for any Tier 2. So car manufacturers do not like to require the use of a specific semiconductor in their specifications (and could thus offer business to Tier 2 suppliers indirectly). Car manufacturers require that certain functions are achieved, but not how it is done.

Hence, the OEM–Tier 2 relationship is not as clearly defined and the levers for a successful one are trickier to pin down. The role of personal relationships and long-term experiences should not be underestimated. Figure 2.20 visualizes the relationships between car manufacturers and the different supplier levels.

One more aspect to discuss is related to the specific structure of the automotive market. Figure 2.21 shows the accumulated market share in automotive, in which the largest vendors are considered first. In 2012, for example, there were 69 car manufacturers who each had more than 1000 cars registered[14] by customers. The cars of these 69 manufacturers spread over 1310 different car models for 144 brands and represented 99.99% of the market. Nevertheless, only five car manufacturers accounted for more than 50% of the cars registered, and another 13 covered the next 40%. This means that the whole market consolidates to relatively few players and is therefore often referred to as an oligopoly (see, e.g., [51]).

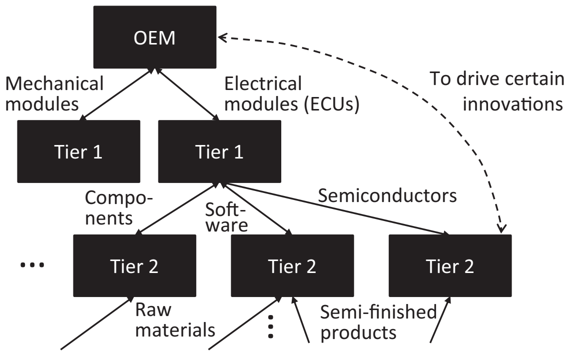

Figure 2.20 Car manufacturer (OEM) and supplier relationships.

Each of the 18 largest car manufacturers is thus a powerful customer to the Tier 1 suppliers; not only because of sheer volume but also because of the type of product and the long product cycles. Traditionally, a car model generally runs for seven years,[15] so once a Tier 1 has passed the hurdle of having their technology designed into a car, it has a long-standing business for a customized product that is not easily replaceable, even in case of an emergency. However, if the technology has not been designed into the car, that particular share of the market is lost for seven years. This results in no insignificant effort by the Tier 1 to make a good impression on the car manufacturers.

2.3.2 Role of the Relationships among Car Manufacturers

Car manufacturers create unique products out of a large number of components. It is not so much each individual component that makes the difference but the right combination of thousands of parts integrated into a specific brand design. The competition, which is fierce also among car manufacturers, focuses on the final product and of course the

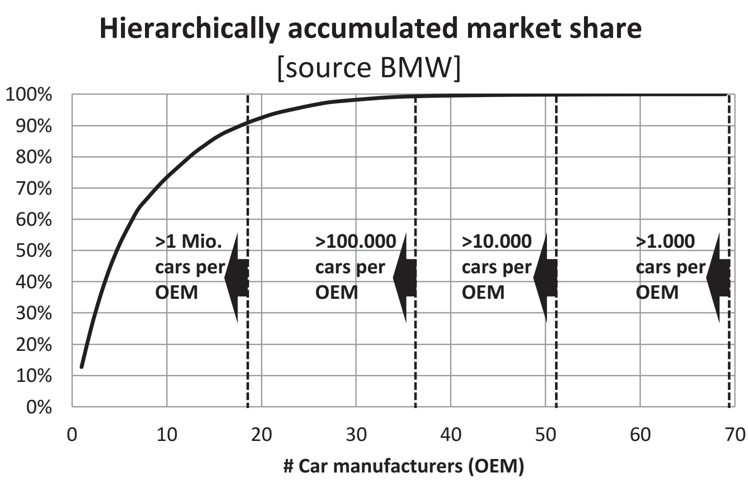

Figure 2.21 Accumulated market share in automotive 2012.

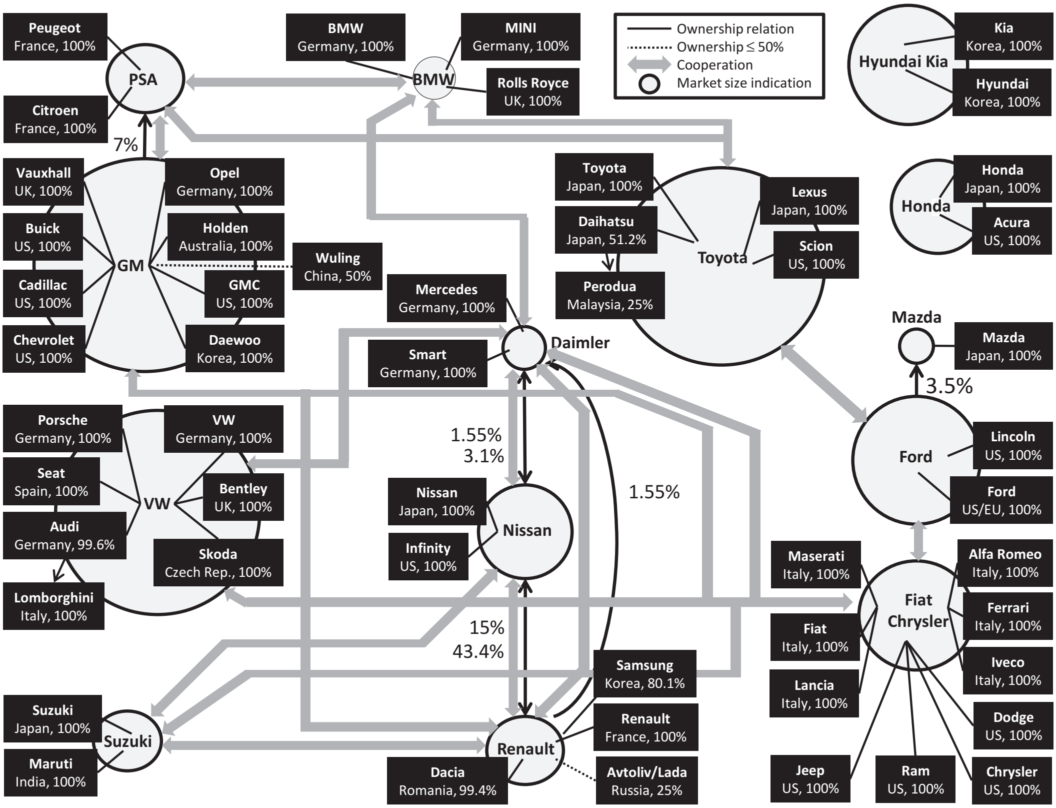

Figure 2.22 Example snapshot of the relationships between the 14 car manufacturers having sold most cars in 2012 with updates on ownership 2016 [54]. Includes information on the size of the manufacturer and which brands belong to it, without brands discontinued in 2013, such as Hummer and Maybach, and brands that sold fewer than 100 cars per year, such as Bugatti.

design, but, with exceptions, not so much on the individual elements. This is particularly true for nondifferentiating elements like in-vehicle networking. Section 2.2 describes some of the communal efforts the car manufacturers made and some of the organizations that were founded in this context. The interactions between car manufacturers nevertheless go way beyond this. Cooperation is found at all levels, from purchasing, through development, to delivering parts to each other, to producing almost the same car. Even though this specific cooperation now seems to be coming to an end, the VW Crafter and Daimler–Sprinter production is a prominent, long-standing example of the latter [52].

Figure 2.22 shows a snapshot of the 2012/2016 interrelations between the 14 car manufacturers who had the most cars registered. Chinese car manufacturers have not been included. Not because of their volumes but because their interrelations especially with non-Chinese car manufacturers have a regulative, i.e., government enforced, edge to it, something which, e.g., blurs the assessment of the Chinese car manufacturers, their car market as well as the relations to other car manufacturers somewhat [53]. Most relationships shown in Figure 2.22 are for pure economic advantages and can change quickly depending on actual ownership changes, trends in the market, and other politics. The diagram emphasizes that in an industry with so much interaction, cooperating in the development of in-vehicle networking or other nondifferentiating functions is a matter of course.

Next to the direct bilateral relationships between car manufacturers, the automotive industry is divided into a multitude of organizations for all different kinds of topics. Depending on the topic and the gain from unification, these organizations are international, or – in a lot of cases – national. It is also not so unusual that a national unity is sought first, before an international unity is attempted. This is not only traditional but often very practical. After all, for the countries in which the car manufacturers are located, the automotive industry often plays a significant role in the national and regional economies, in terms of direct and indirect employment, exports, Gross Domestic Product (GDP), etc. [55]. The industry is often nationally supported and the respective structures have existed for some time. Language barriers and the reduced effort to meet in person add to the seeming nationalism. It is thus not surprising that early in-vehicle networking technologies originated and prevailed in different countries. With the globalization and the omnipresence of digital communication media this is nevertheless changing. The developments around Automotive Ethernet (see Chapter 3) are a good example.

Another important aspect to understand when discussing relations among car manufacturers, and why and how new networking technologies are introduced, is the driving forces behind innovations and their diffusion in the industry. Most innovative functions and features enter the market top down, meaning they are introduced in the high-end car segment first before they are sold in middle class or even small cars.

This has several reasons. First, high-end car customers tend to be early adopters, willing to pay a premium for innovations. It is not unusual that a high-end car is bought fully equipped with all options. Second, it is part of being high end that innovative features are offered in this segment first. Last, but not least, the relative cost of a new feature in a high-end car is significantly smaller than in a small car. It is therefore likely that the economies of scale necessary to allow that feature to be eventually offered in smaller cars, too, are only achieved this way round.

It is the high-end cars and thus the high-end car manufacturers who drive automotive innovations and who bring new features into the industry. The innovation leaders need the support from a more powerful in-vehicle networking system first (see also Section 2.1) and are therefore likely to drive respective developments.

About 10% of the cars produced can be classified "High End" (HE).[16] Table 2.8 lists the ten car manufacturers and brands that sold the highest number of HE cars in 2012, with the manufacturers and brands listed top down according to the number of HE vehicles sold. What can be seen is that those car manufacturers who sell the most HE cars, have neither the highest ratio for HE models to overall models ("HE model ratio"), nor for HE cars sold to cars sold ("HE sales ratio"), nor the highest innovation ranking. Those who do have the highest HE model and HE sales ratio are those for whom HE cars are intrinsic for the company and its existence (BMW AG and Daimler AG). They

Table 2.8 Top 10 high-end (HE) car manufacturers and HE car brands in order of number of HE cars registered, example data from 2012

	HE model ratio (%)	HE sales ratio (%)	Innov. ranking
Car manufacturer			
Toyota Motor Corp.	26	16	5
General Motors Company	26	15	6
Nissan Motor Company	25	21	12
Volkswagen AG	23	8	1
Ford Motor Company	33	13	4
Hyundai Kia Automotive Group	24	10	6
BMW AG	47	32	2
Daimler AG	52	38	3
Chrysler Group LLC	30	26	15
Honda Motor Company	22	11	14
Brand			
Toyota	28	16	4
Nissan	20	20	9
Ford	27	11	3
BMW	50	39	1
Mercedes	52	41	2
Chevrolet	19	12	17
Hyundai	21	13	7
Audi	39	28	8
Honda	17	10	15
Dodge	27	31	20

Note: HE model ratio = number of HE models/overall number of models, HE sales ratio = number of HE cars sold/overall number of cars sold. Innovation ranking of the same year according to [56].
Source: BMW.

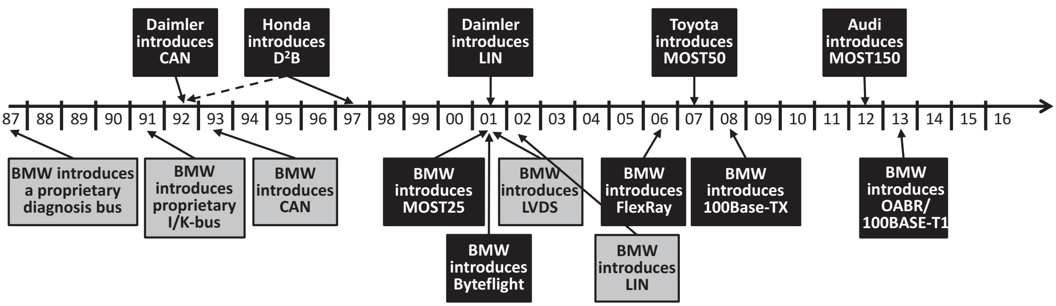

Figure 2.23 Timeline of the introduction of in-vehicle networking systems at BMW and in the industry as such. The event that first introduced a new technology is indicated by a dark box.

also receive top marks when it comes to innovations. In consequence, these companies are likely to experience the limits of existing in-vehicle networking technologies first, and they are also the ones used to drive innovations. Figure 2.2 confirms that European cars have the highest number of networked nodes. It is thus not surprising to see these companies particularly active when it comes to the developments of new in-vehicle networking technologies, as has been described in the subsections of Section 2.2.

Last, but not least, Figure 2.23 shows the timeline in which in-vehicle networking systems have been introduced at BMW and in the car industry as such. It can be seen that BMW, as one of the innovation leaders, often was the first or one of the first to introduce a new in-vehicle networking technology.

Notes

1 Today, there is still a variety of networking technologies inside the cars. A survey performed in 2015 showed that the major car manufacturers use in average 8 different digital communication systems inside their cars today [49]. Additionally, despite the use of digital communication systems, there still is a lot of discrete wiring inside the cars. In one BMW model investigated 50% of the weight of the harness consisted of power supply cables, 43% of discrete wiring and only 7% were used for the bus systems discussed in this book, even though they cover most of the communication. This emphasizes how impossible it would be today, to design a car without in-vehicle networking systems. The vehicle would choke in wiring.

2 The LS CAN transceiver does not only use the differential signal that the HS CAN uses, but additionally evaluates the absolute voltage to ground. In case one wire fails (e.g., breaks or gets disconnected) the LS CAN can theoretically still function in an emergency mode. Nevertheless, this function has not been unambiguously defined for the receiving units and is therefore not really used [57].

3 At the time of writing, the data rate supported by the majority of CAN FD transceivers and the data rate car manufacturers like BMW felt most comfortable with, was 2 Mbps (we deliberately refrain from citations here as there are simply too many parts available and it is not our call to make a selection). However, the standard also addresses 5 Mbps data rate, but does not preclude higher data rates [58].

4 Occasionally CAN is referred to as an "automotive fieldbus." As the terminology originated in industrial control (see also Section 1.2.2) this can only be attributed to its ability to being deployable in almost all areas, i.e., physically spread, locations in the car.

5 The information available on D^2B is not conclusive. What is likely is that the technology had been developed in the late 1970s [59] with the focus on home entertainment and that it was transferred to an IEC standard in the (late) 1980s [60] [61]. Philips definitely played an important role [66], but also Matsuhita [59] [62] and Sony [60] seem to have been involved. For the first in-vehicle use 1992 [60] and, which in the Authors' opinion is more likely, 1997 [63] are quoted, with Honda as the respective car manufacturer [60]. For sure, Daimler introduced D^2B in some of the Daimler models and 1998 is realistic.

6 In the same year the MOST Co was founded, the car manufacturers Chrysler, Daimler, Ford, General Motors, Renault and Toyota also founded the Automotive Multimedia Interface Corporation (AMIC) in order to define a suitable hardware and software interfaces for automotive information, communication and entertainment systems [64]. In this context also FireWire, i.e., IEEE 1394 was discussed as a possible solution. Nevertheless, with the MOST Co gaining more traction and showing faster progress the activity was disbanded in 2004 [65].

7 This is an important primary capability in order to enable autonomous/automated driving.

8 It supports the content protection for Blu-ray, if it is always transmitted over cables in a compressed and encrypted way.

9 Instead of pixel links, they are sometimes also called "SerDes" Interfaces from the Serializer and Deserializer needed to realize the technology (see Figure 2.17).

10 The additional discussion of wireless (consumer) technologies like Bluetooth or WiFi would open up a whole new set of topics to consider. While these are very interesting and being discussed in the industry, we consciously decided not to address these in the context of this book (other than that WiFi would seamlessly integrate into the Ethernet network) in order to concentrate on the aspects important for understanding Automotive Ethernet.

11 In a survey performed within the car industry the participants stated, that while every car manufacturer currently supports in average 8 in-vehicle networking technologies in their cars, the majority preferred between 1 and 4 technologies [49].

12 The V-cycle "Visualizes" the process how and in what order tasks have to be performed in a project. It matches the idea, specification and realization with the respective steps on the test and integration side. The model was first proposed at the end of the 1970s and has become especially popular in automotive (software) development [66] [67].

13 Naturally, in case of malfunctions it is not always obvious whether the Tier 1 has not implemented the communication requirements correctly or whether the car manufacturer has described them insufficiently. With the consequence this can have on the SOP of a car, this is a very serious issue in the industry.

14 Counting "registered cars" results in somewhat different number from looking at the number of cars produced, as not every car built is sold and/or registered (in the same year). The numbers are nevertheless similar enough to make absolutely no differences for the points made.

15 Seven years is a typical model cycle for German car manufacturers. Those seven years generally include a "facelift" in year four to ensure the cars stays up to date with the latest developments. It will be interesting to see whether the car industry can retain this in a world that seems to turn ever faster. There are first indications that model cycles are reduced [46].

16 There are various different ways to classify cars into market segments (see [68] for an overview). In the investigation described here "High End" consists in the classification of the European union of the E-, F, and the upper segment of the J-segments. In the American/British English classifications this translates into: full-size and mid-size luxury cars/executive cars, full-size luxury cars/luxury cars, grand tourers, supercars, and full-size SUV/large 4×4.

References

[1] Brose, "Gründung und Aufbau; 1908–1955," 2012. [Online]. Available: www.brose.com/de/Unternehmen/Historie/1908–1955/. [Accessed 2 August 2016].

[2] J. Donelly, "Needing a Lift, (Maybe) Finding it," 8 August 2008. [Online]. Available: www.hemmings.com/hcc/stories/2008/08/01/hmn_feature9.html. [Accessed 19 August 2013].

[3] Freundeskreis BMW 503, "Die zwei Serien im Vergleich," 2006. [Online]. Available: www.fb503.de/serien.htm. [Accessed 20 August 2013].

[4] T. Streichert and M. Traub, Elektrik/Elektronik-Architekturen im Kraftfahrzeug, Berlin: Springer Vieweg, 2012.

[5] J. Muller, "Another Big Promise by Nissan's Carlos Ghosn: Affordable Autonomous Cars," 27 August 2013. [Online]. Available: www.forbes.com/sites/joannmuller/2013/08/27/another-big-promise-by-nissans-carlos-ghosn-affordable-autonomous-cars/. [Accessed 2 September 2013].

[6] A. Grzemba, MOST, the automotive multimedia network; from MOST25 to MOST150, Poing: Franzis, 2011.
[7] P. Thoma, "Die EE-Geschichte," BMW, München, 1997.
[8] Wikipedia, "On-board diagnostics," 12 July 2016. [Online]. Available: http://en.wikipedia.org/wiki/On_Board_Diagnostics. [Accessed 2 August 2016].
[9] ARC Electronics, "RS232 Tutorial on Data Interface and cables," undated. [Online]. Available: www.arcelect.com/rs232.htm. [Accessed 2 September 2013].
[10] CiA, "CAN Newsletter special issue automotive," October 2006.
[11] Wikipedia, "J1708," 2 February 2015. [Online]. Available: http://en.wikipedia.org/wiki/J1708. [Accessed 2 August 2016].
[12] CiA, "History of CAN technology," 2013. [Online]. Available: www.can-cia.de/can-knowledge/can/can-history/. [Accessed 2 August 2016].
[13] BOSCH, "License Conditions CAN Protocol and CAN FD Protocol," June 2013. [Online]. Available: www.bosch-semiconductors.de/media/automotive_electronics/pdf_2/ipmodules_3/can_protocol_license_1/Bosch_CAN_Protocol_License_Conditions.pdf. [Accessed 23 February 2014].
[14] S. Singer, *per e-mail*, 18 October 2013.
[15] Automotive News, "Top Suppliers," 17 June 2013. [Online]. Available: www.autonews.com/assets/PDF/CA89220617.PDF. [Accessed 24 August 2013].
[16] C&S, "Munich Group Server," C&S, 2013. [Online]. Available: www2.cs-group.de/MunichGroupServer/public/testspec.php. [Accessed 16 October 2013].
[17] BOSCH, "CAN with Flexible Datarate; White Paper Version 1.1," 2011. [Online]. Available: www.bosch-semiconductors.de/media/pdf_1/canliteratur/can_fd.pdf. [Accessed 15 January 2013, no longer available].
[18] BOSCH, "CAN Specification Version 2.0," BOSCH, Stuttgart, 1991.
[19] W. Lawrenz, CAN Controller Area Network, Grundlagen und Praxis, 2nd ed., Heidelberg: Hüthig, 1997.
[20] D. Marsh, "LIN simplifies and standardizes in-vehicle networks," *EDN*, pp. 29–40, 28 April 2005.
[21] LIN Consortium, "LIN Specification Package, Revision 2.2A," LIN Consortium, 2010.
[22] Consortiuminfo, "Local Interconnect Network Consortium (LIN)," 8 July 2013. [Online]. Available: www.consortiuminfo.org/links/linksdetail2.php?ID=428. [Accessed 2 August 2015].
[23] U. v. Burg and M. Kenny, "Sponsors, communities, and standards: Ethernet vs. Token Ring in the local area networking business," *Industry and Innovation*, vol. 10, no. 4, pp. 351–375, December 2003.
[24] H. Schoepp, "In-Vehicle Networking Today and Tomorrow," 17 January 2012. [Online]. Available: www.automotive-eetimes.com/en/in-vehicle-networking-today-and-tomorrow.html?cmp_id=71&news_id=222902008. [Accessed 28 August 2013].
[25] MOST Cooperation, "MOST150 Inauguration in Audi A3," *MOST Informative*, no. 8, p. 2, October 2012.
[26] H. Kaltheuner, "Das Universalnetz, Ethernet-AVB: Echtzeitfähig und Streaming-tauglich," *c't*, no. 13, pp. 176–181, June 2013.
[27] A. Grzemba, *MOST – Das Multimedia-Bussystem für den Einsatz im Automobil*, Poing: Franzis, 2007.
[28] FlexRay Consortium, "FlexRay Communications System Protocol Specification Version 3.0.1," FlexRay Consortium, 2010.

[29] K. R. Avinash, P. Nagaraju, S. Surendra, and S. Shivaprasad, "FlexRay Protocol based an Automotive Application," *International Journal of Emerging Technology and Advanced Engineering*, pp. 50–55, May 2012.

[30] K. Leiß, "Kommunikationssysteme: FlexRay," 26 January 2010. [Online]. Available: www6.in.tum.de/pub/Main/TeachingWs2009Echtzeitsysteme/20100126_FlexRay_K_Leiss.pdf. [Accessed 29 October 2013].

[31] C. Hammerschmidt, "FlexRay not dead, chip vendors claim," 25 October 2012. [Online]. Available: www.automotive-eetimes.com/en/flexray-not-dead-chip-vendors-claim.html?cmp_id=7&news_id=222902569. [Accessed 30 October 2013].

[32] heiseAUTOS, "Schon in einem Jahr kommt ein Infiniti mit elektronischer Lenkung auf den Markt," 17 October 2012. [Online]. Available: www.heise.de/autos/artikel/Nissan-bringt-Steer-by-Wire-in-Serie-1732435.html. [Accessed 20 October 2013].

[33] Wikipedia, "Blu-ray," 25 June 2016. [Online]. Available: https://en.wikipedia.org/wiki/Blu-ray. [Accessed 4 July 2016].

[34] Wikipedia, "Display resolution," 30 June 2016. [Online]. Available: https://en.wikipedia.org/wiki/Display_resolution. [Accessed 4 July 2016].

[35] ISO, "ISO/DIS 17215 Road Vehicles – Video Communication Interface for Cameras (VCIC) parts 1–4," ISO, Geneva, 2013.

[36] Blu-ray.com, "Blu-ray FAQ, How Fast Can You Read/Write Data on a Blu-ray Disc?," 2016. [Online]. Available: www.blu-ray.com/faq/#bluray. [Accessed 4 July 2016].

[37] Wikipedia, "Low-Voltage Differential Signaling," 26 May 2016. [Online]. Available: http://en.wikipedia.org/wiki/LVDS. [Accessed 2 August 2016].

[38] Texas Instruments, "LVDS Owner's Manual," 2008. [Online]. Available: www.ti.com/lit/ml/snla187/snla187.pdf. [Accessed 29 August 2013].

[39] Wikipedia, "FPD-Link," 16 June 2016. [Online]. Available: http://en.wikipedia.org/wiki/FPD-Link. [Accessed 2 August 2016].

[40] Maxim, "Gigabit Multimedia Serial Link with Spread Spectrum and Full-Duplex Control Channel," October 2014. [Online]. Available: http://datasheets.maximintegrated.com/en/ds/MAX9259-MAX9260.pdf. [Accessed 26 February 2017].

[41] J. Schyma, "Big Data Rates in the Car: Ethernet & More Over One Single Link System," in *5th Ethernet&IP@Automotive Technology Day*, Yokohama, 2015.

[42] Kraftfahrt-Bundesamt, "Fahrzeugzulassungen (FZ) Bestand an Kraftfahrzeugen und Fahrzeuganhängern nach Fahrzeugalter 1. Januar 2013," January 2015. [Online]. Available: www.kba.de/SharedDocs/Publikationen/DE/Statistik/Fahrzeuge/FZ/2015/fz15_2015_pdf.pdf?__blob=publicationFile&v=3. [Accessed 3 August 2016].

[43] bookofjoe, "Microsoft Windows Automotive – World's First Production Car with a Factory-Installed USB Port," 12 March 2006. [Online]. Available: www.bookofjoe.com/2006/03/microsoft_windo.html. [Accessed 15 September 2013].

[44] just-auto, "USA: Cars to Get USB ports Soon – CD Players to Become Obsolete?," 31 October 2005. [Online]. Available: www.just-auto.com/news/cars-to-get-usb-ports-soon-cd-players-to-become-obsolete_id84660.aspx. [Accessed 15 September 2013].

[45] R. Sheldon, "USB Communications Introduction," 2008. [Online]. Available: www.controlanything.com/Relay/Device/A0003. [Accessed 2 September 2013].

[46] T. Grünweg, "Modellzyklen der Automobilherstelle: Eine Industrie kommt auf Speed," 10 February 2013. [Online]. Available: www.spiegel.de/auto/aktuell/warum-lange-entwicklungszyklen-fuer-autohersteller-zum-problem-werden-a-881990.html. [Accessed 15 April 2013].

[47] Automobil Konstruktion, "Marktforderung: modularer Mehrfachnutzen," February 2010. [Online]. Available: www.autokon.de/weiterbildung/-/article/16537511/26315427. [Accessed 11 March 2013].

[48] G. Volpato, "The OEM-FTS Relationship in Automotive Industry," *International Journal of Automotive Technology and Management*, vol. 2, no. 3, pp. 166–197, 2004.

[49] K. Matheus, "The Future of In-Vehicle Networking – Survey Results," February 2016. [Online]. Available: http://hanser-automotive.de/fileadmin/heftarchiv/bmw/FA_BMW_ONLINE_LANG.pdf. [Accessed 27 February 2017].

[50] K. Sievers, "Semiconductors – Enablers of Future Mobility Concepts," 22 February 2012. [Online]. Available: www.nxp.com/wcm_documents/news/download-media/presentations/nxp-sievers-zvei-elektromobilia-final.pdf. [Accessed 16 August 2013].

[51] J. Bouman, "Section 3: Characteristics of an Oligopoly Industry," Inflate your mind, 2011. [Online]. Available: www.inflateyourmind.com/index.php?option=com_content&view=article&id=134&Itemid=165. [Accessed 17 August 2013].

[52] Auto.de, "Mercedes Sprinter/VW Crafter: Die Kooperation wackelt," 20 October 2012. [Online]. Available: www.auto.de/magazin/showArticle/article/88559/Mercedes-Sprinter-VW-Crafter-Die-Kooperation-wackelt. [Accessed 11 March 2013].

[53] M. Grzanna, "China's Love-Hate Relationship With The German Automotive," 24 April 2013. [Online]. Available: http://international.sueddeutsche.de/post/48759305591/chinas-love-hate-relationship-with-the-german. [Accessed 17 August 2013].

[54] Volkswagen Konzernkommunikation, "Wer mit wem? Verflechtungen in der Autobranche," May 2012. [Online]. Available: http://viavision.org/index.rnd?id=54. [Accessed 11 March 2013].

[55] OECD, "Chapter 2: The Automobile Industry in and beyond the Crisis," 2009. [Online]. Available: www.oecd.org/eco/outlook/44089863.pdf. [Accessed 17 August 2013].

[56] Center of Automotive Management, "Automotive INNOVATIONS Award 2012," 2 May 2013. [Online]. Available: www.auto-institut.de. [Accessed 9 May 2013, no longer available].

[57] S. Meier, "CAN, Controller Area Network," 2013. [Online]. Available: www.siegfriedmeier.de/Download/X/CAN.pdf. [Accessed 16 October 2013].

[58] ISO, "ISO/DIS 11898–2, Road Vehicles – Controller Area Network (CAN) – Part 2: High-Speed Medium Access Unit," ISO, Switzerland, 2015.

[59] Wikipedia, "D²B," 11 July 2013. [Online]. Available: http://de.wikipedia.org/wiki/D2B. [Accessed 17 September 2013].

[60] M. Pöllhuber, "Plastic Optical Fiber," 2001. [Online]. Available: http://tkhf.adaxas.net/cd1/Optik.pdf. [Accessed 17 September 2013].

[61] Wikipedia, "IEC 61030," 16 July 2016. [Online]. Available: http://en.wikipedia.org/wiki/IEC_61030. [Accessed 3 August 2016].

[62] IT Wissen, "D2B (Domestic Data Bus)," undated. [Online]. Available: www.itwissen.info/definition/lexikon/domestic-data-bus-D2B-D2B-Bus.html. [Accessed 17 September 2013].

[63] Tyco Electronics (TE Connectivity), "High Speed Data Networking for the Automotive Market," 2007. [Online]. Available: http://docplayer.net/19160927-High-speed-data-networking-for-the-automotive-market.html. [Accessed 2 August 2016].

[64] PRNewswire, "Automotive Multimedia Interface Collaboration Announces Formalization of Objectives and Operating Procedures," 27 April 1998. [Online]. Available: www.prnewswire.com/news-releases/automotive-multimedia-interface-collaboration-announces-formalization-of-objectives-and-operating-procedures-74245312.html. [Accessed 17 September 2013].

[65] Wikipedia, “Car Classification,” 1 August 2016. [Online]. Available: http://en.wikipedia.org/wiki/Car_classification. [Accessed 3 August 2016].

[66] Die Beauftragte der Bundesregierung für Informationstechnik, “Häufg gestellte Fragen zum V-Model XT,” Die Beauftragte der Bundesregierung für Informationstechnik, 2012. [Online]. Available: www.cio.bund.de/DE/Architekturen-und-Standards/V-Modell-XT/Haeufig-gestellte-Fragen/haeufig_gestellte_fragen_node.html;jsessionid=D2388DE9513E3BDA57ABA3C398DD4088.2_cid289#doc2157266bodyText1. [Accessed 11 March 2013].

[67] J. Schäuffele and T. Zurawka, Automotive Software Engineering – Grundlagen, Prozesse, Methoden und Werkzeuge, Wiesbaden: Vieweg + Teubner, 2005.

[68] The Hansen Report, “AMI-C Nearly Spent,” Hansen, Portsmouth, 2004.

3 A Brief History of Automotive Ethernet

When a new technology is being developed and enabled in an industry, there are various factors that impact the success of that technology. In the authors' opinion the most important ones are its benefits, its costs, and the framework that allows an industry to develop around it. This chapter will discuss these topics with respect to Automotive Ethernet. However, as is also frequently pointed out (see, e.g., [1] [2] [3]), it is not only technical facts but also individuals who act as the driving force behind a new technology. In respect to Automotive Ethernet, both authors feel that they have their share in the events. In consequence, the descriptions in this chapter will sometimes reflect personal viewpoints.

3.1 The First Use Case: Programming and Software Updates

3.1.1 Architectural Challenges

In 2004, BMW decided to introduce a central gateway ECU in its cars starting from 2008 Start of Production (SOP) onwards. This central gateway was to combine two functions: (1) to route data between the different CAN, FlexRay, and MOST busses inside the cars; and (2) to function as the diagnostic and programming interface with the outside world. For the latter, BMW has always used a centralized approach. This means that software can only be flashed with an external tester device that is connected via the On-Board Diagnostics (OBD) connector [4] and that in case flashing is performed, all flashable Electronic Control Units (ECUs) inside the car are updated with their newest software versions. This approach assures that the customer always has the latest software in the car and that there are no sudden software inconsistencies in functional domains, in which some units have been updated and others have not. This is an architectural choice that, as it happens, was decisive for the introduction of Automotive Ethernet. Nevertheless, there are car manufacturers that use decentralized approaches for software updates and handle the version management differently. They might update, e.g., the multimedia/infotainment unit only and use USB or DVD for it, while using the OBD connection to update other, individual ECUs.

In 2004, BMW used a High-Speed CAN (HS CAN) interface with the OBD connector for connecting the tester to the in-vehicle network. The physical limit of the HS CAN

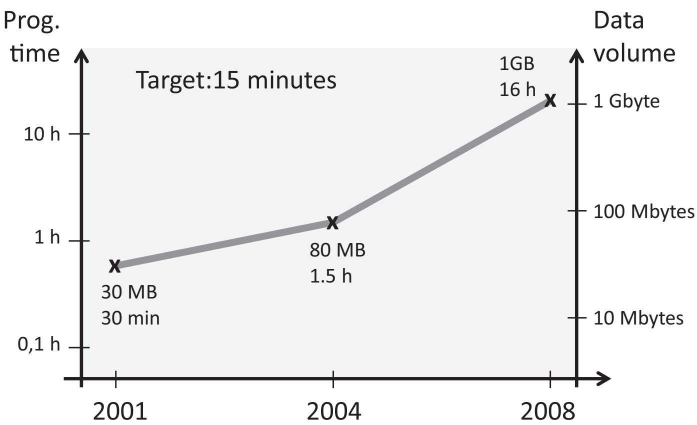

Figure 3.1 Flash data volume and programming time predicted in 2004 for 2008 at BMW [5].

was/is 500 kbps (see also Section 2.2.2); the additional overhead due to the protocols needed for this application reduced the net data rate to about 200 kbps. The prognosis for the accumulated amount of flash data in 2008 exceeded 1 Gbyte; some of the multimedia devices had several 100 Mbyte of software (including map data), and also the new FlexRay connected devices in the chassis domain needed a significant amount of software. Taking everything into account, the complete software update of a well-equipped, high-end car would have exceeded 16 hours. At a dealer, this would have meant sending customers home with a replacement car and making them come back the next day, even though all they potentially required was an update of the map data. In the factory a flash time of 16 hours was unthinkable. The target duration for the software update was set to 15 minutes (see also Figure 3.1). Obviously, another interface technology than HS CAN was needed.

3.1.2 Potential Car Interface Technologies

The new technology needed to fulfill several basic requirements:

1 The data rate had to be sufficiently high. The intention was to flash all units/busses connected to the central gateway in parallel. The flash memory in the Head Unit (HU) at the time allowed for write access at 15 Mbps. Additionally, a FlexRay bus with 4 Mbps, 2 HS CAN busses, and 1 LS CAN had to be serviced. This led to a required net data rate of about 20 Mbps from the interface.
2 The flash process is performed only a very few times during the lifetime of a car. It was therefore not acceptable to require additional processing resources in the central gateway for the flash process only. It was thus important that the selected interface technology would not overstress the available resources.
3 It was intended to have the flash process as part of a (world wide) networking function. Within the local garage, the network allows to have more than one external tester connected to the car at a time, or to have one tester connected to multiple cars. On a

worldwide scale, with the latest software on a central server, a good integration into the network would allow this software to be flashed directly into a car at any dealer in the world.

4 The solution needed to be cost efficient, both inside the car and in the test equipment used at the dealer and factory. BMW intended to introduce a new flash concept. Backward compatibility with existing systems was not required.

3.1.2.1 Evaluation of MOST

In principle, MOST provided a higher data rate than was needed and it had been introduced in BMW serial cars three years earlier, in 2001. Nevertheless, MOST 25 was also considered unsuitable. The primary use case for MOST 25 was synchronous audio communication. In respect to the required high-speed data communication, this had the following disadvantages:

- **Insufficient data rate:** The maximum net bandwidth on the MOST 25 asynchronous data channel is only about 7 Mbps (the gross data rate of 12.7 Mbps mentioned in Section 2.2.4.2 reduces further owing to protocol overhead).
- **High resource demand:** To achieve the maximum net bandwidth of 7 Mbps, it is necessary to use data packets of 1014 bytes, i.e., ~1 kbyte. Additionally, it requires using a block acknowledgment for 64 packets that is part of the so-called MOST-high protocol. For the software update use case this would have meant completing the reception of 64 MOST packets before being able to send them on. In consequence, 64 kbyte RAM would have been needed for this procedure only. Additionally, the block acknowledge would have affected the routing between MOST 25 and the other bus systems, which alone was estimated to take up the computation power of a complete Central Processing Unit (CPU).
- **Wrong topology:** It is in the nature of a tester that it is only temporarily attached to the car. Because MOST requires a ring topology, this would have either meant adding a second MOST ring between tester and gateway during testing, or extending the ring when the tester was attached. Both concepts were unattractive for complexity reasons.
- **No IP support:** MOST 25 did/does not speak IP, i.e., does not provide for routers, switches, or even hubs. To integrate the system into the diagnostics network at a BMW dealer or BMW as such would have required significant effort and workarounds. IP support was only introduced later with MOST 150. MOST 25 relied on the MOST-high protocol only.
- **New interface:** MOST would have been a completely new interface for the external testers that are yet developed and used with different technical background and focus. Adding MOST to the testers, would have required adding the respective hardware and software interfaces to the testers along with the introduction of the communication paradigms of MOST to the diagnostic application.
- **High costs:** The interface is comparably costly.

It was discussed in 2004 that a next generation MOST would be developed and that it would support IP and data communication better. Some of the disadvantages thus might have been easier to overcome. However, this was too far out and too immature to base a decision on. In consequence, BMW decided against using MOST as the diagnostic car interface; and rightly so. MOST 150 saw market introduction in 2012 only (see also Figure 2.23).

With a gross data rate of 10 Mbps, the data rate FlexRay provided was obviously too small. Also, in 2004, FlexRay had not yet been introduced, and as its SOP was planned for 2006 this raised concerns about the maturity of the technology.

3.1.2.2 Evaluation of USB

The next interface investigated was USB 2.0. USB was well known as a consumer interface and was on the roadmap of many car manufacturers to be introduced as an interface for consumer devices (see also Section 2.2.7). It was very common in the PC environment and thus suitable for the external testers as well. Also, with 480 Mbps data rate, the bandwidth of USB 2.0 was more than sufficient. Nevertheless, when investigating USB in detail, the following disadvantages led to the decision not to use USB as the diagnostic interface:

- **Insufficient robustness/immunity:**[1] To achieve sufficient signal integrity, expensive cables and connectors would have been needed with the use of USB.
- **Insufficient cable length:** USB allows only for a cable length of about four meters, which is a disadvantage in a large garage.
- **No network support:** As said, the idea was to have more than one external tester connected to the car at a time, or to have one tester connected to more than one car. With USB this would have led to a collision of multiple USB controllers, or to very complex, nonstandard compliant workarounds.
- **New protocol:** The automotive protocol stack and driver had to be developed for the use case.

In consequence, USB was also not the right solution. LVDS/pixel links were never investigated, because they could not support networking (see also Section 2.2.6). FireWire alias IEEE 1394 had disadvantages similar to those of USB. Additionally, the physical interface was unclear and the automotive industry had not yet collected any experience with the technology.

3.1.3 The Solution: 100BASE-TX Ethernet

The next technology to investigate was Ethernet. It provided a sufficient data rate, was readily available in computers and laptops, and was a networking technology, i.e., it promised to fulfill the idea of handling the car as a node in the world wide network and in a larger (diagnostic) network at the dealer. In this network, multiple cars are connected to one tester or multiple testers are connected to one car, while all being connected to the backend at BMW. In 2004, the idea of using Ethernet in automotive was unheard of. But, Ethernet was/is a well-documented technology and provides a good infrastructure, so it was possible at BMW to assess its suitability.

3.1.3.1 The Physical Layer

The element that was expected to be most critical for Ethernet was the Physical Layer. The anticipation was that, as for USB, the automotive robustness requirements would result in (too) expensive cabling and connectors inside the car. This turned out to be wrong. The first experiments in the ElectroMagnetic Compatibility (EMC) lab were run with two PCs connected via two pairs of simple Unshielded Twisted Pair (UTP) CAN cables that BMW had used in serial production for some years. These cables did not at all comply with the standard CAT 5 cable defined for 100BASE-TX Ethernet. Yet, the very first measurement results for the immunity showed that the setup met the immunity requirements for in-vehicle communication, without any modifications being necessary!

The situation was different for the EMC emissions.[2] The emissions were way beyond the limit lines and would have caused audible reception distortions in the FM radio, if used at runtime.[3] Nevertheless, in the case of software updates at the dealer or in the factory the car is stationary and not in runtime, i.e., by a customer driving and listening to the radio. To ensure that the 100BASE-TX UTP Ethernet connection could not cause distortions during runtime, BMW added an "activation line" in the later implementation. This activation line ensured that the 100BASE-TX UTP Ethernet connection inside the car would be active only when the external tester was connected.

To start with, it was expected that RJ-45 connectors had to be reused, and all the first investigations used these. In the end, this turned out to be unnecessary. It was possible to add the two wire pairs necessary for the 100BASE-TX connection to the OBD connector (see Figure 3.2). The well-established and standardized connector offers four vehicle manufacturer specific pins, and BMW decided to use those for the 100BASE-TX connection. Measurements in the EMC lab proved that the immunity still met the requirements. So, after the initial evaluation, in 2005, 100BASE-TX was considered a promising technology and a decision was taken to seriously investigate its use.

3.1.3.2 Protocol Stack and Software

With the CAN interface, BMW used the Unified Diagnostic Services (UDS) protocol. UDS describes the handling of diagnostic information in automotive and is specified in ISO 14229–1 [7]. When moving to a new networking technology, BMW wanted to avoid defining a new protocol and new sequences for the software update, even though BMW did not require backward compatibility to the existing implementation. At the same time, Ethernet was an "IT Technology" with a pool of available protocols and technologies. Thus, the next step after establishing the principle suitability of the PHY was to investigate the reuse and adaptability of standard IT protocols for Ethernet-based diagnostic communication in automotive.

Table 3.1 shows the result. It is special, because it showed for the first time how IT and automotive standard protocols can be matched. It was the first time at BMW that no new protocol had to be developed from scratch for automotive use – instead the focus was on reuse and synergies. With only a small addition, called "High-Speed Fahrzeug Zugang" (HSFZ), which was needed in order to enable the parallel flash process and to map the UDS onto TCP, it proved to be perfectly possible.

Table 3.1 Comparison of OBD-Stacks, migrating from HS CAN to Ethernet

OBD over CAN	OBD over Ethernet	Interface
UDS	UDS	The same diagnostic devices and the same protocol
n/a	HSFZ	Maps UDS onto TCP and organizes parallel flashing
CAN transport protocol	TCP/IPv4	IPv4 and TCP used instead of the CAN transport protocol
CAN controller	Ethernet MAC	Use of the Ethernet MAC instead of the CAN controller
CAN transceiver	100BASE-TX PHY	CAN transceiver changed for a 100BASE-TX Ethernet PHY
Two pins at OBD interface	Four pins at OBD interface	Ethernet 100BASE-TX uses four pins instead of the two used for CAN

At the same time as collecting the protocols and solutions, their portability from Linux[4]) to automotive operating systems needed to be investigated. For multimedia ECUs this would not have been so much of an issue, as they are normally based on modern operating systems like Linux or QNX.[5] The gateway, however, was a typical automotive ECU using an OSEK[6] Operating System (OS) with much lower memory

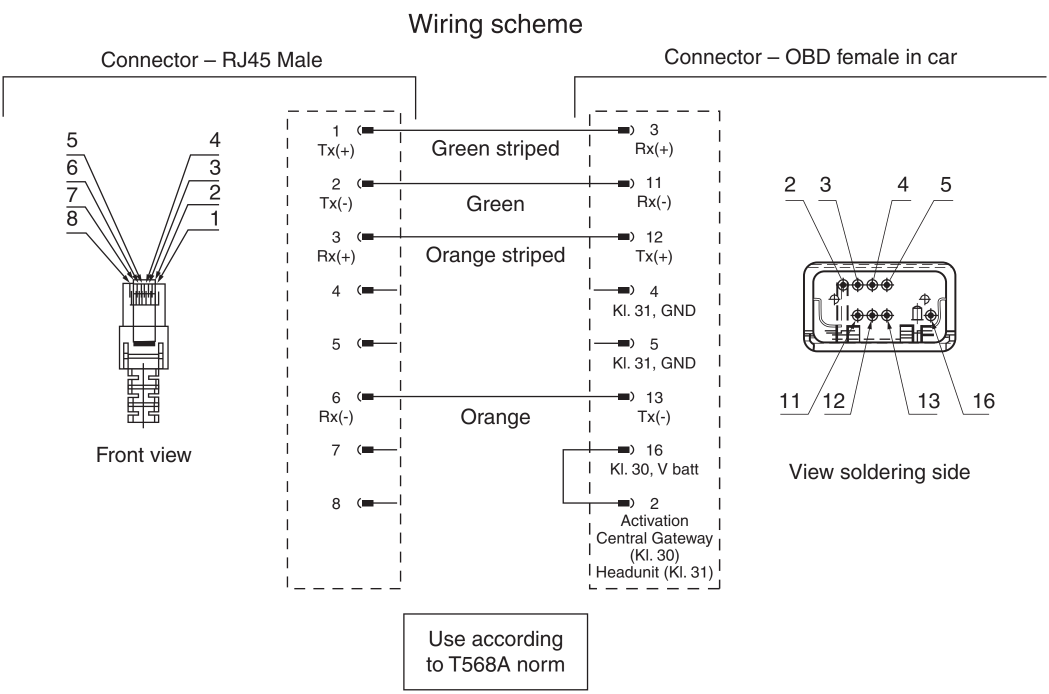

Figure 3.2 RJ-45 100BASE-TX Ethernet connector in relation to the vehicle OBD connector [6].

resources. The question was whether it would be possible to have a suitable software stack on such a typical automotive ECU without overstressing the available resources. It was possible, and all needed functionality was implemented in the central gateway within the given resource bounds.

As said, implementing an Ethernet-based communication system had not been done before at BMW, nor in the automotive industry as such. Hence, there was a significant amount of skepticism and anxiousness about the feasibility. Yet, the results were impressive. The gateway project included also the implementation of CAN, FlexRay, and MOST 25; busses onto which the data being flashed into the car via Ethernet had to be distributed. The Ethernet implementation, despite being new, caused the fewest error tickets during the qualification of the ECU compared with the implementations of other networking technologies supported. Furthermore, with Ethernet, it was for the first time possible to use freeware software stacks in the development process; one example being the "lightweight IP stack" [8]. This helped tremendously to prove that Ethernet was doable at a point in time, when no one would have dared to invest heavily into the solution. Available test specifications and programs as well as existing test infrastructure for interoperability tests were an additional bonus.

3.1.3.3 The Car as a Node in the Network

With using Ethernet as the car interface, i.e., with the respective SOP in 2008, it was possible to treat the car as a network node connected to the external world, i.e., the dealer's or BMW's network. Figure 3.3 visualizes this. In the example "*n*" cars are connected to various utilities like testers, programming devices or the server using an external, standalone switch. One car can be connected to various utilities and one utility can be connected to various cars at the same time. The Ethernet interface thus provides more than just a high-speed link to the car.

In the depicted use case, IPv4 addresses are assigned to the cars by a Dynamic Host Configuration Protocol (DHCP) server. With the help of the unique Vehicle Identification Number (VIN) each car is unambiguously identified and in combination with the temporary, but also unique IP address the car can be located in any workshop around the world. The diagnostic application in the external equipment communicates via the UDS protocol to the car's internal devices. The car internal switched architecture – see the example of car "*n*" in Figure 3.3 – provides for this. With using Ethernet, the test software can be installed on normal PCs, instead of needing proprietary hardware, which is another benefit.

The described BMW efforts are a BMW specific solution. Nevertheless, in 1998 the United Nations initiated the World Wide Harmonised On-Board Diagnostics (WWH-OBD) effort, with the goal of a harmonized standard for emissions control [9]. In this context, IP was selected as the communication protocol between on-board and off-board diagnostic applications [10]. The resulting Diagnosis-over-IP (DoIP) ISO 13400 standard is (and this is not accidental) based on the same principles as the BMW solution: It enables the UDS applications via TCP/IP and a 100BASE-TX Ethernet interface [11].

Even if it was for diagnostic and flash purposes only, car "*n*" in Figure 3.3 indicates that the central gateway as well as the Head Unit (HU) use an internal Ethernet switch.

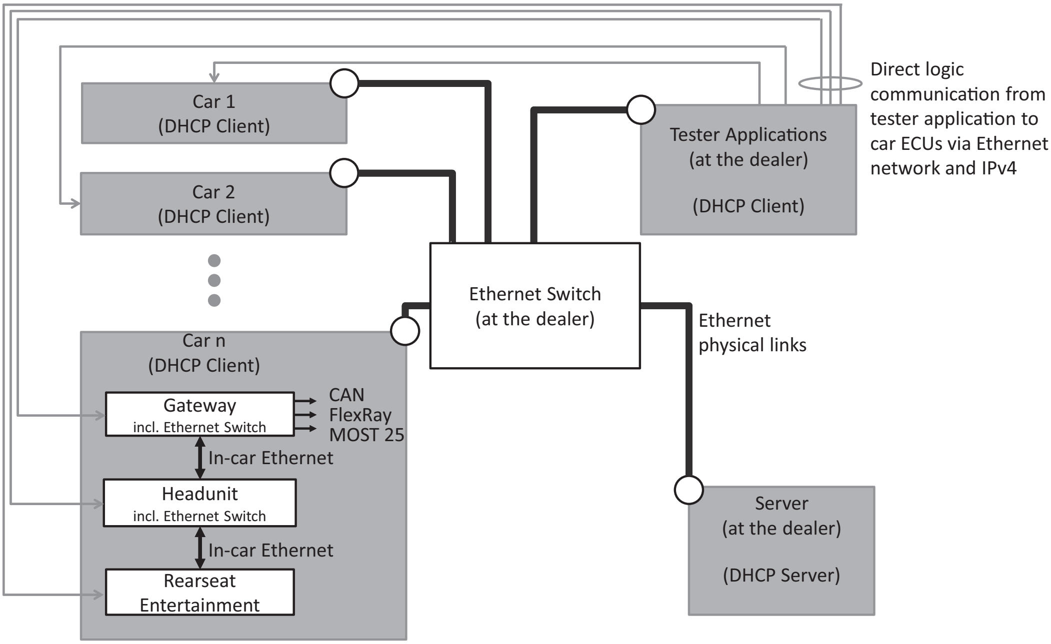

Figure 3.3 The car as a network node.

BMW started with a small network and limited use cases, but it provided an excellent learning base and allowed the derivation of guidelines for future in-vehicle high-speed networks. Note, this was developed in 2005/2006. In other words, it took about 10 years from the first assessments to rolling out Ethernet as an extensive system bus in BMW cars in 2015 [12].

3.1.3.4 Automotive Semiconductors for 100BASE-TX

One last aspect needed to be solved: the availability of automotive suitable Ethernet chipsets. Various vendors sold and still sell 100BASE-TX PHYs and switches, but automotive has severe requirements that need to be fulfilled in order for semiconductors to be used inside vehicles. On a broad scale these are [13]:

- Cost effectiveness
- Fast start-up
- Reliability
- Long-term maintainability
- Scalability and flexibility (in case of extras/options)
- Suitability for critical environmental conditions (temperature, vibration, humidity)
- High EMC fitness
- Low weight
- Small size
- Low power consumption

Some of the above requirements are technology dependent (e.g., scalability), some depend on the willingness of semiconductor suppliers (e.g., long-term maintainability), and some depend on their capabilities (all quality related aspects as well as size, weight, power consumption etc.). In 2005, when BMW started looking for parts to use in production, Ethernet in automotive was a completely new idea. In general this meant that the traditional automotive semiconductor suppliers did not sell Ethernet chips and that the traditional Ethernet suppliers did not consider the automotive market to be particularly promising in order to justify investing into the automotive qualification.

In the end, it turned out that Micrel (now Microchip), who was selling a similar portfolio into industrial automation, was interested. In a joint effort between Micrel and BMW a qualification plan was devised. BMW benefited in two ways from this approach:

1 BMW had direct knowledge of potential risks and weaknesses and was able to set up appropriate actions in parallel with the qualification of the semiconductors in order to ensure the SOP in 2008.
2 BMW was able to gather experience with handling semiconductors that had no base in the automotive ecosystem. This allowed BMW to amend the qualification program accordingly and to learn for the future growth of Ethernet in automotive.

The results of the qualification program were good. Only the housing of the chips had to be changed in order for the chips to pass the tests for ElectroStatic Discharge (ESD).[7] For the diagnostic application under discussion, electromagnetic emissions were not an issue, so it was not necessary to investigate this aspect. With the diagnostic interface having been introduced consecutively in all BMWs since the SOP in 2008, Micrel (and now Microchip) has made a good choice and can now – at a time when the success of Ethernet in automotive is no longer questioned – rightfully claim to have been the first company with AEC-Q100[8]) qualified Ethernet products [14]. For BMW, the target of needing only 15 minutes for flash updates was met in a close enough proximity to call the project a full success.

3.2 The Second Use Case: A "Private" Application Link

In parallel to the programming and diagnostic use case described in the previous section, a special use case was planned for the 2008 high-end Rear Seat Entertainment (RSE). This application was to reuse the navigation data stored in the Head Unit (HU) in the RSE and required that the navigation data could be transmitted from the HU with about 20 Mbps. MOST 150 was not available at the time and MOST 25 did not accommodate 20 Mbps for data communication, as discussed. So, together with Harman Becker (now Harman International), the supplier of the respective HU and RSE at the time, it was decided to use 100BASE-TX Ethernet for the communication between the units. As the link was a private link between two units of the same supplier, the development was the responsibility of the supplier, who chose a QNX-based implementation.

Other than for the programming use case described in Section 3.1, this link was to be used during the runtime of the car and ElectroMagnetic Emissions (EME) did make

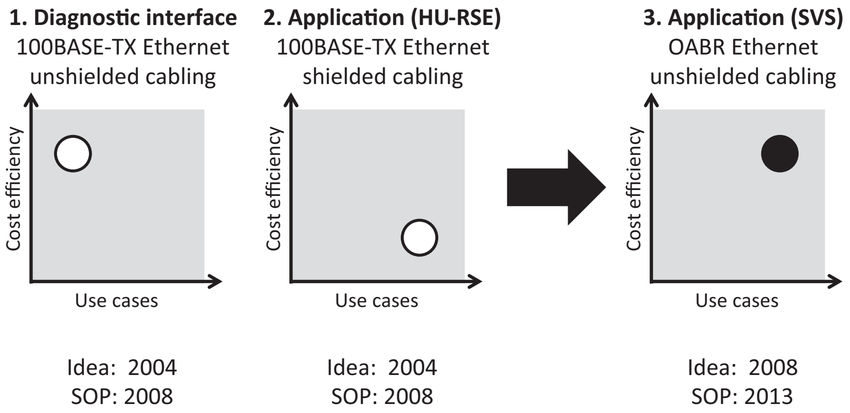

Figure 3.4 Limitations of 100BASE-TX Ethernet at BMW in 2008 and target achieved with OABR/100BASE-T1 in the first application, the Surround View System (SVS) [24] [5].

a difference. Consequently, the cabling for the HU–RSE link required shielding, which made it heavier, more expensive, and less attractive.

With the harness in the car being the third heaviest and the third most expensive component in the car [15], the weight and costs of any connection inside the car are of importance. Also, while economies of scale and cost reductions are expected for semiconductors, cabling does not comply with the same market mechanism. The price for copper is very volatile [16]. This means that it is unrealistic to expect a cost reduction in cabling in the same way as for semiconductors.

So in 2007,[9] BMW was at a turning point. Ethernet looked promising, but was not quite there yet. As visualized in Figure 3.4, using 100BASE-TX as a PHY technology either meant a restriction of use cases or not being competitive costwise. It required the discovery of Unshielded Twisted Single Pair (UTSP) Ethernet (also called BroadR-Reach, OPEN Alliance BroadR-Reach (OABR) or now 100BASE-T1)[10] to make Ethernet attractive for automotive (as will be described in the following sections). In 2007, before the discovery of 100BASE-T1, the situation was totally different. BMW had been one of the founding members of the MOST Corporation and the first car maker to introduce that technology. Using Ethernet with shielded cabling had no cost advantage over MOST. Also, MOST had been developed for streaming audio, and seemed much more suitable for high-quality customer experiences than best-effort Ethernet.

In consequence, some engineers thought it would be more useful to enhance MOST with better data and IP capabilities than to invest into Ethernet; the results of which can be found in MOST 150. Nevertheless, BMW also started several research programs on the use of Ethernet and IP in automotive [17]. Their early focus was on Quality of Service (QoS) and timing behavior. This coincided with the efforts at IEEE, where the Audio Video Bridging standardization projects had been started [18] and interesting material was available. The BMW activities yielded good results (see, e.g., [19] [20] [21] [22] [23]) and led also to the start of a project funded by the German government called SEIS (Security in Embedded IP-based Systems) in 2009.[11] Among other aspects,

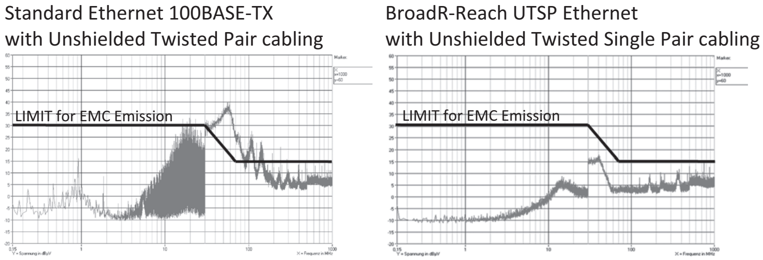

Figure 3.5 World's first automotive measurements of the ElectroMagnetic Emissions (EME) of an Ethernet BroadR-Reach link, performed in January 2008. The results even exceeded the performance of many traditional networking technologies.

setting up and pursuing this project served to create a community in the German automotive industry for Automotive Ethernet. However, the PHY remained the bottleneck.

3.3 The Breakthrough: UTSP Ethernet for Automotive

BMW decided to synchronize all hitherto knowledge with the future requirements on IP-based communication systems. A key learning was that a PHY usable with unshielded cabling was decisive for the future of Ethernet in automotive. Another was that the EMC properties, at least the immunity, had been surprisingly good the first time round. In consequence, BMW decided to look more closely at the possibilities to reduce the emissions of 100BASE-TX Ethernet when using unshielded cabling.

Together with Lear, who had supplied the central gateway, BMW performed measurements. Starting point was the existing gateway. The EMC performance of the ECU itself, when not doing Ethernet transmission, was very good. The hope was thus to be able to isolate the source(s) for the strong emissions. And indeed, the output driver stage was identified as the root cause. The gateway hardware allowed the output driver of the Ethernet PHY to be deactivated, while leaving the internal MII and all other interfaces live. In this case, the unit was well under the emission limit lines. Unfortunately, irrespective of the means taken – filters, ferrite beads, etc. – it was not possible to get below the emission limit lines when the output drive stage was switched on.

Thus, in summer 2007, BMW approached four well-known vendors of Ethernet PHYs and asked for their opinions and solutions. Colleagues at the automotive semiconductor supplier Freescale (now NXP) had suggested that this might be worthwhile. Of the companies addressed, only Broadcom responded positively, and in September 2007 the first meeting was held in Munich. During this meeting the results of the gateway measurements were discussed and the automotive requirements were aligned with the performance value of a solution Broadcom had originally developed for Ethernet in the First Mile (EFM). In January 2008, the Broadcom technology called BroadR-Reach went into the EMC labs at BMW. Figure 3.5 shows the results of the first emission measurements performed.

Of the other three companies, one did not reply and the other two thought the BMW request impossible. Some years later, after BMW and Broadcom had proved that transmitting Ethernet packets at 100 Mbps over unshielded cabling was possible in the automotive environment and were promoting BroadR-Reach in order to attract other customers and suppliers, every one of the three other companies originally asked, developed other, incompatible solutions. One solution was even based on 100BASE-TX (see Section 4.3.1.3), something BMW would have highly appreciated a few years earlier.

On one hand these solutions created confidence in the industry that transmitting 100 Mbps Ethernet packet over unshielded cabling in the automotive environment is really feasible. One the other hand, for those not having yet decided on the use of BroadR-Reach, it caused additional validation and decision effort and some uncertainty for all. In a fragmented market, no one wants to have decided for the technology with the smaller and potentially decreasing market share.

In hindsight we know that BroadR-Reach/100BASE-T1 succeeded, but at the time the situation was not always that clear. All car manufacturers have a long lead time to introduce new in-vehicle networking technologies. The decision has to be taken at least three years ahead of SOP, meaning that another year before investigations on the technology have to have started. For BMW all other proposals were simply too late to consider. This meant that for BMW the other solutions were mainly a source of discomfort as they posed an economical risk.

With BroadR-Reach, the door opener to Automotive Ethernet was found. BroadR-Reach promised to transmit Ethernet packets at 100 Mbps at vehicle runtime over a single pair (100BASE-TX requires two) of unshielded cabling (Unshielded Twisted Single Pair, UTSP), i.e., the same cabling the industry used for CAN or FlexRay networks. This would be the most cost-efficient high-speed network in automotive, providing a higher data rate than MOST at a lower price level than MOST, any pixel link, or consumer technology. Nevertheless, this was still only the beginning. In 2008 all that existed was a good technical prototype some engineers at BMW had had the chance to investigate. The technical and economic feasibility had yet to be proven over all levels of decision making within BMW. Also, the automotive industry as such had yet to be convinced that Automotive Ethernet was the right way forward. After all, the use of an in-vehicle networking technology has limited advantages to a car manufacturer when the car manufacturer is the only one using it (see also Section 2.4).

3.4 BMW Internal Acceptance of UTSP Ethernet

3.4.1 Yet Another In-Vehicle Networking Technology

BMW was one of the first car manufacturers to introduce in-vehicle networking as such and one of the first to introduce CAN and LIN. BMW was a founding member of the LIN, FlexRay, and MOST consortia and the first car manufacturer to introduce MOST 25, FlexRay, and 100BASE-TX Ethernet in serial production cars (see also Section 2.2). The company had especially invested in the MOST technology, and built up know-how

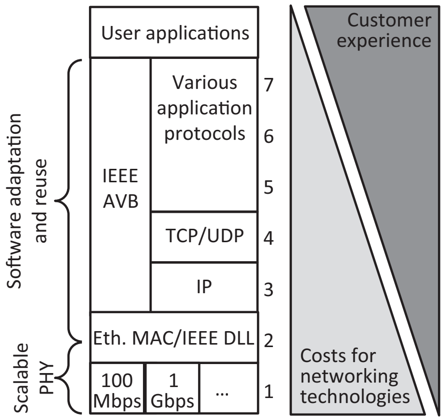

Figure 3.6 Flexibility, scalability, and reuse in Automotive Ethernet.

and experts. Additionally, MOST 150 was going to offer a higher data rate than MOST 25 as well as better data/IP support. So, why adopt yet another networking technology?

It is true that BMW has invested a lot in in-vehicle networking technologies in the past. BMW is one of the innovation leaders in the industry (see also Section 2.4) and therefore always one of the first car manufacturers to need new in-vehicle networking technologies with different properties. After all, the in-vehicle networking provides an essential infrastructure for distributed applications. At the same time, having worked with all the networking systems means to have accumulated significant networking know-how, to have observed the increase in complexity in the systems, and to realize that to constantly completely change technologies is not sustainable in the long run. A more future-proof system was needed that is flexible, that scales, and that allows for reuse. The bandwidth requirement in cars is expected to continue to increase and it is no longer acceptable to constantly change the technology because of it.

Ethernet-based/IP-based in-vehicle networking provides all of this (see Figure 3.6). As it builds on the ISO/OSI layering model, changing to a higher data rate requires first changing the Physical Layer (PHY) technology only, while from the Data Link Layer (DLL) upwards the software can potentially be reused. It is also possible to use a different medium, e.g., like wireless or optical, without many changes. If a new protocol needs to be added on the application layers, this can be added without touching the layers below. Ethernet will eventually allow a reduction in the number of networking technologies as well as the resources bound by them. Instead those resources will be able to focus on innovations with direct customer use.

At a higher level, Ethernet-based communication also addresses a general challenge the automotive industry faces: the ever increasing product differentiation combined with the trend toward shorter model and innovation cycles [25]. Car manufacturers handle this by modularization and building block systems that allow designers to compose certain domains of a new car from sets of building blocks. The in-vehicle network has to support this. As was explained above, Ethernet-based communication provides for scalability in respect to data rates and transmission media. Additionally, a switched

Ethernet network adds new possibilities and flexibility to the networking design [26]. A switched network can have all kinds of topologies and is not restricted to a ring or line. Increasing or reducing the number of ECUs is significantly simplified (see also Section 6.3.2.2). Furthermore, Ethernet offers the possibility to separate networks virtually with the help of Virtual LANs (VLANs) even if they use the same physical network (see Section 5.2).

So, in principle, it was understood that Ethernet-based in-vehicle networking was the right way forward. The question was, how to introduce it on a larger scale into the vehicle? The application area that presented itself for the introduction was the infotainment domain. The first calculations that compared MOST 25 with shielded 100BASE-TX Ethernet however did not yield any obvious cost advantage. In the end, how do you quantify "future-proof"? BroadR-Reach Ethernet was too new. The first measurement results were promising, but many voices also within BMW doubted that UTSP cabling would really work. On top the infotainment domain was/is seen as one of the keys for the customer experience of a car. The existing MOST solution had a well-established, automotive experienced supplier base. Ethernet, at that time, did not. Clearly, a different pilot application was needed in order to prove the feasibility, strength, and maturity of Automotive Ethernet and the BroadR-Reach technology.

3.4.2 A Suitable Pilot Application

BMW chose to use the Surround View System (SVS) as a pilot application for BroadR-Reach/100BASE-T1 Ethernet. The purpose of a SVS is to show the surroundings of a car when it is being parked and in the following it is explained why the SVS was particularly suitable. The existing SVS system was already using digital LVDS/pixel links to transport the uncompressed data streams of each individual camera to the ECU for generating the surround view picture. This surround view picture was sent "ready to be displayed" to the Head Unit (HU) via an analog FBAS connection (see also Figure 3.7), while the HU sent its control data to the SVS via CAN. The SVS controlled the cameras via LIN. For risk minimization reasons in the pilot application, only the pixel links and the LIN control links were replaced by UTSP Ethernet, while the connection between SVS ECU and HU remained the same. As the Ethernet links provide with 100 Mbps a much smaller data rate than the pixel links even in 2009 – and a smaller data rate than the new High-Definition (HD) imager would generate – the video streams from the cameras needed to be compressed.

From the application point of view this raised two concerns: first, whether the loss of information caused by the compression would impair the performance of the image processing algorithms, and second, whether the latency introduced by compression and decompression would be acceptable. An early prototype that included the use of an Ethernet link with compression and decompression had been set up by the research department. The results were encouraging and the investigations were subsequently sufficiently refined to remove any concerns on the feasibility (see, e.g., [27] [28] [29]). Concerning the latency H.264 and Motion JPEG (MJPEG) were investigated. Not all modes of H.264 were suitable; those suitable were not available in hardware at the time of

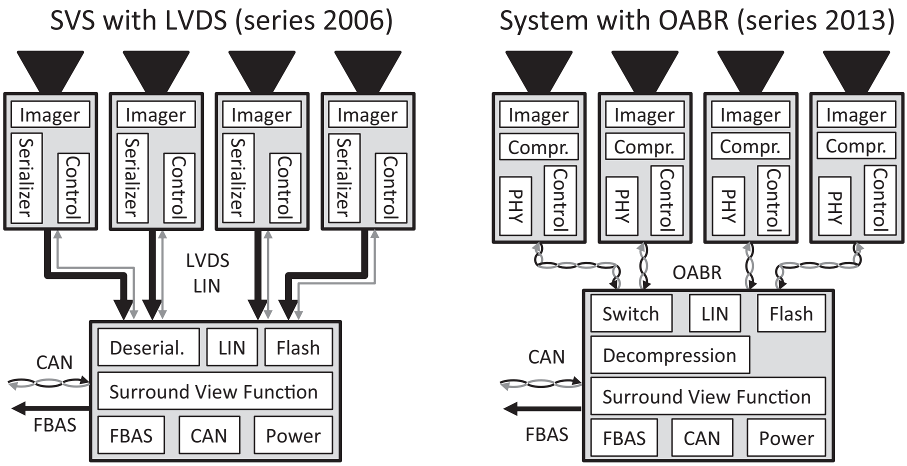

Figure 3.7 The pilot surround view system application [31].

investigation, though they yielded good results in simulation. In the end, it was the joint effort with the μC supplier Freescale (now NXP), which resulted in a product allowing for a low latency implementation using MJPEG compression, which sufficiently addressed the original concerns.

So the SVS was selected as the pilot application. It turned out to serve as an optimal pilot use case for several reasons:

1 It held the **right technical challenges.** The main focus was on proving that the **EMC** requirements could be met using **UTSP cabling** also in a real life application. This included the selection of **standard cables and connectors**, the choice and development of a μC with low power dissipation and low EMC emissions [30], the decision on the transformers/common mode chokes (**CMC**)/filtering to use (or not to use), and the investigation of the influence of temperature changes (for more details see Chapter 4). As the spatial constraints of a camera are particularly tight, a camera can be seen as a worst-case use case in respect to thermal influences and operating temperature. The small size of the camera was also be a challenge in terms of software, which needed to reuse as much of available IT technology (see also Chapter 5) while at the same time it had to be portable onto the **small embedded controller** available [30]. Last, but not least, the **automotive qualification** of all previously nonautomotive parts, like the BroadR-Reach semiconductors, had to be achieved.

2 It had an **excellent business case.** The cameras providing the respective images in a SVS need to be located in the extremities of the car. In consequence, the cables leading there are long, some pass through several inline connections and some end in wet areas, i.e., cables and connectors need to be water-resilient. Shielded Twisted Pair (STP) cabling as well as the respective shielded and partially waterproof connectors in small spaces result in significant costs. The Ethernet system required some more effort in the cameras due to the compression, but the savings in the harness more than outweighed these extra costs [30]. In fact, the OABR/100BASE-T1 Ethernet

technology was the first high-performance networking technology that financed its introduction by what it saved, including interest. This is extremely unusual but also very helpful.[12]

3 It was a **low-risk** application. In the first step, only the pixel links between cameras and SVS ECU were exchanged (see also Figure 3.7). The link between the SVS ECU and the HU stayed the same. This means that while offering a good business case and relevant technical challenges, the new Ethernet links did not impact the communication inside the rest of the car. In the worst case, there would have been a fall back. Note that the SVS generation following the here described pilot migrated also the SVS to HU connection to Ethernet.

4 It had **optimal timing.** The target SOP in 2013 meant that the SVS and the Ethernet connections were being developed two years ahead of the next new 7-series BMW with SOP in 2015. New functions and innovations are generally introduced top down. This meant that for the 2015 7-series BMW a more extended Ethernet in-vehicle network was of interest, so the proof of the network usability had to be provided in sufficient time before. The same introduction concept had successfully been used with FlexRay, so it was seen as the right way to proceed with Ethernet, too.

5 It **proved the commitment.** Some additional risk was seen in working with suppliers inexperienced in automotive. After all, automotive has a long return on investment period. It often takes four to five years after semiconductors have been developed, before the first cent comes rolling back. Especially for companies who are based in the consumer industry, this is completely unheard of. Additionally, each car model is produced for about seven years and might need replacement parts for another 20 years. So, the car manufacturer has to trust not only the technical solution but also in the long-term commitment of the semiconductor supplier. The supplier can prove the commitment with a local support network, product roadmaps, etc. The pilot project offered a comfortable time window in which new suppliers were able to familiarize themselves as well as comply with the necessities of the automotive industry.

3.4.3 The Future of Automotive Ethernet at BMW

The fulfillment of technical requirements is generally not sufficient for deciding on a technology. For example, the technology also needs to be affordable, for which a promising business case always provides a strong argument. Nevertheless, in an environment with limited resources like the engineering workforce of a company, these have to be used wisely. Even if there is money to be saved in one case, maybe it is (even) better for a company to use the same resources in another project. Thus, also the long-term implications of the decision have to be taken into account. In the case of Automotive Ethernet it provides technical solutions for an otherwise unsustainable situation. Also the suppliers showed commitment to the automotive market. Despite all this, it additionally needs to be possible that a market can develop around the technology. The related aspects – multi-sourcing, future developments, Tier 1 suppliers and other car manufacturers, etc. – were essential for the BMW internal decision, too. As those aspects will be discussed in the following Sections 3.5 and 3.6, this section concentrates on the elements relevant for BMW.

The first EMC measurements with the BroadR-Reach technology were performed at BMW at the beginning of 2008, the decision to use the technology for the pilot application was taken in March 2010 and the SOP of the respective Surround View System (SVS) was in September 2013 [32]. This means that BMW decided to investigate the technology thoroughly during the world economic crisis of 2008 and 2009, in order to be able to decide on series production in March 2010. At a time when many predevelopment projects in the industry were stalled for lack of funding, BMW allocated money and engineering power to Automotive Ethernet, which was thought technically undoable at the time even by many players in the Ethernet industry.

The obvious explanation is that Automotive Ethernet had a strong case, technically as well as financially. In the authors' opinion this is not sufficient though. In the authors' opinion the spirit, which makes BMW one of the most innovative car manufacturers [33], has its share; not only because a powerful in-vehicle networking system is an enabler for innovations. It is part of being an innovation leader to dare to go into unknown terrain while at the same time being able to assess the risk correctly and being able to handle the challenges. No innovation leader would be innovation leader without this being part of the company's culture. It implies motivated engineers and a capable management, too.

During the preparation of the pilot project decision and in the first year after, many important technical questions and challenges were addressed (for the results see also Chapters 4–6). From the nucleus of the project group the knowledge on the achievements was passed onto larger groups within the company (and to the outside world, see Section 3.6). Personal networks and selected partners in, e.g., qualification or research, who generally have a good exposure to management, helped spreading the knowledge to a critical mass. It is the car manufacturer who is responsible for the in-vehicle network (see also Section 2.3). Decisions as consequential as using Automotive Ethernet as the basis for a large-scale network inside the car, require a strong base for acceptance; over all involved departments and hierarchy levels. When taking decisions of this scale, not everything can be expected to run smoothly. When problems arise, the engineers need to want to overcome the hurdles and the management needs to want to back them up. After all, there is a social component in all technical developments.

The results of these efforts were successful. In March 2011, BMW took the decision to migrate the infotainment domain from MOST 25 to 100BASE-T1 Ethernet instead of to MOST 150, with target SOP in 2015. It was a goal of BMW to digitize all video streams inside the car. The existing MOST 25 system did not provide sufficient bandwidth for this, so a migration to a new system was necessary. BroadR-Reach/100BASE-T1 Ethernet was the more cost-efficient solution. In October 2011 the decision followed to migrate part of the driver assist domain, also starting in 2015. In this case more bandwidth was needed for new innovations and the integration of Ethernet seemed more future-proof than to add yet another CAN or FlexRay or two of them.

3.5 The Industry Framework for a New Technology

"The discussions regarding standard adoption are technical, but it is people and firms that must agree to the standards. Not surprisingly, this means that there is a social component to this process" [2]. Adoption of a new concept, standard, or technology is multifaceted. Not only do technical and economic questions need to be answered. A framework needs to be in place that serves as a breeding ground in which the new technology can thrive. The more suppliers expect to profit from a new standard the more likely it is going to succeed [34]. Additionally, there are individual preferences, animosities, and paradigms. These influence the decision processes, but they themselves are influenced by the availability of structure that allows for an industry to develop as well.

3.5.1 From a Proprietary Solution to an Open Standard

The BroadR-Reach technology was developed by Broadcom, who also owns the respective Intellectual Property Rights (IPR) on the technology and its trademark. This means that to start with, the technology was proprietary,[13] i.e., closed to competitors. Closed technologies lead to monopolies and these are undesirable. Not only can it be expected that the prices the customers pay in a monopoly situation are unfavorable, also the customer depends on one supplier for reliability, availability, future developments, and innovations. Products in a competitive situation are simply better for the customer and generally the industry as such.

Fundamental for not getting into a monopoly situation and for allowing intrastandard competition is that the IPR holder embraces this. There are numerous examples, of which Ethernet itself is actually one, that show how even an allegedly inferior technology can win over a superior technology. This is simply because the IPR holder of the inferior technology pursues a truly open licensing policy, while the technology owner of the superior technology does not or does so only half-heartedly. According to [2], IBM, in the hope of a market advantage or unawareness, lost the whole LAN market from Token Ring to Ethernet, because of IBM's inconsequential technology opening, even though they did offer it for standardization in IEEE 802.5.

In the end, there are three possible ways to make a technology be an open standard (see also Table 3.2 and note 11): (1) the standard is offered to a Standard Setting Organization (SSO) for publication; (2) the standard is published via a Special Interest Group (SIG)/industry consortia; and (3) the standard is simply published directly by the IPR holder. Provided the IPR holder executes a Reasonable and Non-Discriminatory (RAND) licensing policy, all three ways are viable and have been chosen successfully in the past [34] [35]. While the first is probably the most accepted, it bears the risk of delays and the risk of changes to the original technology, unless the original technology has already been successfully established as a de-facto standard. The third option is the least transparent and the success relies very much on the IPR holder. If it is mainly

Table 3.2 Options on how to open a technology and their consequences [5]

	More suppliers	Acceptance	Influence on technology	Timing
Give the standard to an SSO	Interest depends on	Good, well-established, and transparent	A not yet established technology is likely to change	Complete process >3 years
Create a SIG that publishes the standard	Market prospect: A promising market finds interested suppliers	Transparent, but not established at the beginning	Technology does not need to change, control shifts to SIG	Faster, depends on founders
Leave it to the IPR holder		Not transparent, very dependent on the IPR holder	Only IPR holder defines technology	Fast

the customer that requests the opening of the technology, this is not a good start – full support of the IPR holder is a fundamental requirement for the success – and can lead to misunderstandings.

In the case of Automotive Ethernet, companies from two industries with different cultures had to rely on each other, while at the same time timing was crucial. In consequence, the second option was chosen. In November 2011, NXP, Broadcom, and BMW started the One Pair EtherNet (OPEN) Alliance.[14]) The companies that joined within November 2011 were C&S, Freescale (now NXP), Harman, Hyundai, Jaguar Landrover, and University of New Hampshire Interoperability Lab (UNH-IOL) [35].[15] As it happened, the OPEN Alliance became one of the fastest growing automotive consortia and announced more than 300 members in March 2016 [36].

The overall goal of the OPEN Alliance was to help establish Ethernet-based communication as an in-vehicle networking technology. An early focus was on the 100 Mbps BroadR-Reach/100BASE-T1 technology. In order to support other semiconductor vendors in developing competitive BroadR-Reach products, the specification was reviewed, clarified, and enhanced. Functional as well as EMC compliance tests were defined and interoperability tests were developed. After all, the OPEN Alliance had the right members, with the UNH-IOL the entity in compliance and interoperability tests in the Ethernet world, with C&S a known entity in the automotive certification world for traditional in-vehicle networking systems and FTZ with knowledge on EMC testing (see also Section 3.6). Other test houses followed. For usability by, e.g., the Tier 1s and car manufacturers, OPEN specified the components (cable, connectors, harness manufacturing) and identified suitable tools.

Multiple sources are essential for a market to prosper and the prognosis of a market to prosper is essential for multiple sources to be offered. The mentioned activities of the OPEN Alliance aimed at achieving more planning security for semiconductor and other vendors wanting to enter and invest into the market. Not only were technical risks reduced by OPEN, e.g., to end up with a noninteroperable solution, the members of OPEN also represent the interest of the market. Additionally, OPEN was set up to address any issue that hampers the adoption of Automotive Ethernet, either by finding/defining a solution within OPEN or by cooperating with other organizations better suitable to take up the task at hand. In consequence, the technical work has grown also. The OPEN Alliance started with five technical committees. At the time of writing technical committee 12 ("1000BASE-T1 Interoperability and Compliance Tests") had just started, whereas other have completed their work and are in hibernation [37].

One last aspect to discuss in the context is the difference between intra- and interstandard competition. The worst thing that can happen to a customer is to be faced with a monopoly that leaves no alternatives. This is nevertheless very rare. If there is a promising market and one company has a proprietary solution for this market but is not going to license it, generally other, technically different solutions will be created by competitors seeing an opportunity in the same market [38]. This leads to a market with interstandard competition. If the product is a standalone product, i.e., it requires no complementary products, no minimum distribution, or no interoperability of any kind, the customers have healthy competition despite the fact that the solutions technologically differ. In the case of communication technologies, however, interoperability, standardization, and network effects[16] are key. The more manufacturers produce products with/for/of the same technology the better for the customer. Technologies with interstandard competition can work to some extent – in automotive pixel links is an example – but generally speaking interstandard competition slows the market development of communication technologies down.

Interstandard competition leads to insecurity among the customers [39]. No one wants to have invested into a technology that does not succeed and that in consequence might be discontinued. Some customers in such situations even delay their decision to adopt a technology. This in return leads to smaller volumes and reduced economies of scales, which again makes the whole market less attractive. If there is no intrastandard competition, i.e., a number of competitors offering products that are interoperable, interstandard competition is better than no competition at all. If there is a chance for intrastandard competition, interstandard generally competition harms the development of the industry.

In the authors' opinion the OPEN Alliance played a strong role (next to the IEEE adoption of the BroadR-Reach standard discussed in the next section) in aligning the market to a single solution and ensuring that – despite other solutions being proposed – the market became an interstandard competition market.

3.5.2 Shaping the Future at IEEE

Ethernet-based communication is attractive because it potentially scales, i.e., the MAC and software layers can stay the same, while the PHY is being replaced with one that supports higher data rates. The IEEE has a Gbps PHY technology for copper wiring available, but 1000BASE-T requires four pairs of twisted cables and was expected to additionally need shielding if used in the automotive environment. This meant that, while in principle Ethernet provided the possibility to scale to a higher data rate in the

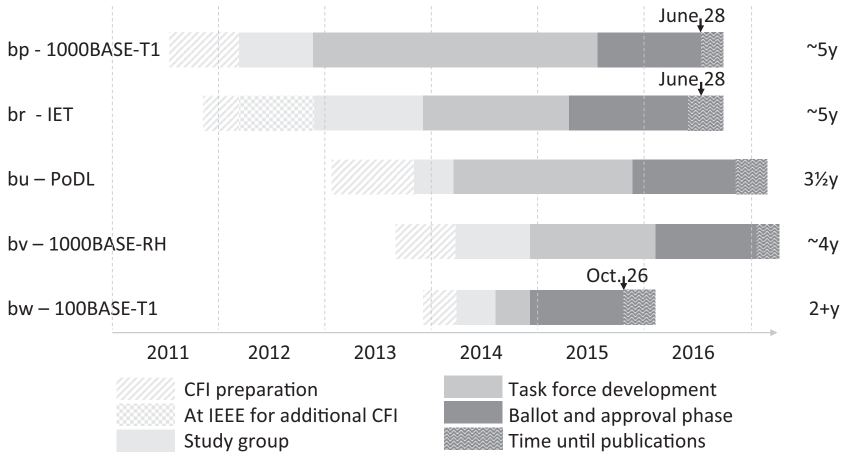

Figure 3.8 Timeline of IEEE 802.3 standards in the realm of Automotive Ethernet (almost) concluded by the end of 2016 [42].

automotive environment, in practice a suitable technology had yet to be developed. So not only the present, but also the future of Automotive Ethernet had to be initiated.

At BMW it was estimated that Gbps Ethernet would be needed for serial production starting from 2018. In order to achieve this, the final decision would have to have been taken by 2015, with a prior chance to evaluate the technology. So, after the first critical milestones for 100 Mbps had been met, efforts started in the middle of 2011 to standardize an automotive suitable Gbps Ethernet at the IEEE in order to meet this timeline. In March 2012 the Call for Interest (CFI) for the "Reduced Twisted Pair Gigabit Ethernet" (RTPGE) passed [15] and a respective IEEE study group was established, which was successfully turned into the task force IEEE 802.3bp by the end of 2012 [40]. In January 2014 the task force agreed on renaming RTPGE to 1000BASE-T1 and IEEE concluded the standard in June 2016 (see Section 4.3.2.1 for technical details).

Unfortunately this was longer than expected and too late for 2018 SOP. But, independent from the details of the standardization process, an important structural step had been achieved with starting 1000BASE-T1 at IEEE. IEEE 802.3 represents the home of Ethernet, with the respective experts and interested industry representatives present. With 1000BASE-T1, automotive was established as a new application field in IEEE 802.3 and with that opened a path for the future. Figure 3.8 shows the other standardization efforts useful for automotive that had followed. In July 2013 the CFI for 1 Pair Power over Data Line (1PPoDL) passed [41] and the respective task force IEEE 802.3bu was established by the end of the same year [40]. PoDL refers to a concept in which power is transmitted over the cables that are also and originally used for data transfer. In the Automotive Ethernet context, this is particularly attractive for sensors, like cameras, that are located in the extremities of the cars. PoDL therefore allows a reduction in the number of cables and hence a reduction in the harness weight and volume. The IEEE had standardized Power over Ethernet (PoE) first in 2003 for the two pair 100BASE-TX, which, of course, is not suitable in the case of a one pair Ethernet variant (see Section 4.3.3 for more details).[17]

Figure 3.8 shows that also 1000BASE-RH was initiated as an optical 1 Gbps solution that could serve the automotive industry as well as the home and other markets and that, last but not least, BroadR-Reach was rubberstamped as 100BASE-T1. Despite this being the latest standard to have been initiated in this first round it needed the shortest time, with only one meeting cycle as study group and one as task force before the document moved into ballot phase. The Interspersing Express Traffic (IET)/IEEE 802.3br standard is somewhat an outsider. First, it was initiated by Industrial Automation (albeit potentially useful for automotive) and second, it serves to increase the efficiency of the channel while reducing some latencies; something that is discussed with Time-Sensitive Networking (TSN) in Section 5.1.4. At the time of writing new automotive speed grades were being discussed for starting standardization efforts in IEEE 802.3: Multiple Gbps Ethernet and 10 Mbps Ethernet [43] (see also Section 4.3.3).

3.5.3 Supportive Structures and Organizations

Ethernet-based communication in automotive, or Automotive Ethernet, is not only about the Physical Layer. It covers all layers of the ISO/OSI layering model (see also Section 1.2.5). One of the prime attractions of using Ethernet-based communication is the opportunity for reuse. This applies to technical solutions as well as to organizations developing these. However, reuse and adaptations from the IT industry are only one side, integrating Ethernet-based communication into various existing automotive efforts is the other. The following list gives a brief overview of the main organizations and activities BMW engaged with other than IEEE and OPEN in order to establish Ethernet-based communication in automotive: AUTOSAR, AVnu, GENIVI, and the ISO 17215 standardization of a Video Communication Interface for Cameras (VCIC).

- In 2003, as the amount and complexity of software in automotive was continuously increasing, key players of the industry launched the AUTomotive Open System ARchitecture (**AUTOSAR**) development partnership [44]. The main goal was to enable the exchange and update of software and hardware over the service life of a vehicle. For this, AUTOSAR developed a software architecture standard that also has to cover the communication interfaces. The "AUTOSAR Operating System" (OS) was designed to be suitable for many types of applications and today is an integral part of many ECUs. It was therefore fundamental for the introduction of Ethernet-based communication in automotive that AUTOSAR supported the respective protocols. To start with, AUTOSAR 4.0, which was published at the end of 2009, provided means to support Diagnosis-over-IP (DoIP), i.e., Ethernet communication based diagnosis and software flashing via IP and UDP [45]. Since then the Ethernet capabilities have continuously increased with the consecutive AUTOSAR versions: Version 4.1 (2014) added, e.g., TCP, Service Discovery (SD), and the connection to the MAC and PHY layers [46]; Version 4.2 (also 2014) optimizes the resources [47]; Version 4.3 details the support of Ethernet switches.

- The **AVnu Alliance** was founded in August 2009 [48] with the goal of promoting the emerging IEEE 802.1 Audio Video Bridging (AVB) and the related IEEE 1722 and 1733 standards (see Sections 5.1.2 and 5.1.3 for the technical description). These IEEE standards enhance Ethernet-based communication systems with Quality of Service (QoS) functionalities. However, the standards offer various choices. AVnu set out to overcome potential ambiguities with profiles, certification, and plug fests. AVnu focused originally on the three application areas professional audio, mobile devices, and automotive [49] (industrial was added as a fourth application field later [50]). In order to support and guide the ongoing standardization activities around TSN with harmonized automotive requirements (see also Section 5.1.4) AVnu, e.g., established the so-called "Avnu sponsored Automotive AVB gen 2 Council" (AAA2C) [51] [52]. From day one, AVnu promoted the use of Ethernet in automotive and the availability of Audio/Video QoS for respective applications. With the lack of QoS being seen as a major flaw of Ethernet in comparison with, e.g., MOST, AVnu thus provided an important contribution to the cause. AVnu published the automotive profile [53] in order to simplify the qualification process for Ethernet-based communication systems in the automotive industry.
- When a car manufacturer, e.g., buys a Surround View System (SVS) from a supplier today, it buys the cameras and the control ECU from the same supplier. This is not always the optimal solution as a supplier who is good at image processing is not necessarily good at building cameras. In order to allow for buying cameras separately from the ECUs the automotive industry initiated in 2009 a standardization activity at ISO: **ISO 17215**, **Road vehicles** – **Video Communication Interface for Cameras** (VCIC). At that time BMW was at the beginning of the surround view project. Various technologies were being discussed for the networking technology to use. In the end, Ethernet-based communication succeeded and BroadR-Reach was recommended for the Physical Layer technology [54].

  Note that in addition to the early ISO efforts of VCIC, in 2016 ISO started the project **ISO 21111, Road vehicles** – **In-vehicle Ethernet**. This project was initiated by Japanese industry players in order to support the deployment of optical Gigabit Ethernet (see also Section 4.3.2.2) in the vehicle [55] and the project was originally named "Road vehicles – In-vehicle Gigabit Ethernet" [56] [57]. However, ISO agreed later in 2016 to rename and restructure the project such that it can comprise all specifications needed to enable Automotive Ethernet and not provided elsewhere, independent of speed grade and medium. At the time of writing the standardization activity had just started and no finalized specifications where yet available.
- The **GENIVI** Alliance was founded also in 2009 with the goal of driving the broad adoption of an in-vehicle open source development platform [58]. The idea was to spur software development and to achieve shorter product life cycles by collaborating on a common, Linux-based reference platform and by fostering an open source development community. The GENIVI platform includes Linux-based services, middleware, and open application layer interfaces. In consequence, the principles intended to be used for the communication middleware of Automotive Ethernet had to be integrated into GENIVI. The Scalable service-Oriented MiddlewarE over IP (SOME/IP,

see also Section 5.4) was thus made available as a GENIVI library. GENIVI is a good example on how diverse the activities are that all need to be taken into account, when introducing a new networking technology in automotive.

3.6 Industry-Wide Acceptance of Ethernet

The historic development of in-vehicle networking technologies had taught the industry that such nondifferentiating functionalities are more beneficial if broadly accepted and widely deployed in the industry (see also Chapter 2). A high probability of industry-wide acceptance was therefore required also inside BMW to move ahead. Both the adoption by Tier 1 suppliers as well as the adoption by other car manufacturers was and is relevant. Tier 1 suppliers function as multipliers of new technologies. The car manufacturers are their customers who might request those technologies or who adopt the new technologies because the Tier 1 offers them at good value.

To assure that a Tier 1 supplier is interested in a new technology, a car manufacturer can simply ask for it. But for a Tier 1 to really embrace that technology the Tier 1 needs to experience or at least expect many car manufacturers to be interested. For the success of Automotive Ethernet, it was therefore important to convince other car manufacturers, i.e., the competitors, on the advantages of Ethernet-based communication. How do you convince someone you do not have a business relationship with? In the end, every car manufacturer has to evaluate the benefits and economic impacts internally, in line with their key market segments and other economic considerations.

Nevertheless, this can be supported. Other car manufacturers had similar EMC requirements and could be expected to pay similar prices for components as BMW did. Thus, expecting interest, BMW pursued a proactive information policy and actively approached competitors, as well as Tier 1 and Tier 2 suppliers. Also, every company interested on their own accord was welcome to discuss Ethernet-based communication. BMW encouraged competitors to perform their own EMC measurements, in order to reduce skepticism on the feasibility. After all, seeing is believing. These efforts had two important results:

1 The **inclusion of independent organizations.** The University of Applied Science in Zwickau (FTZ) got involved. As an independent entity FTZ had performed EMC tests on in-vehicle networking technologies for the automotive industry in the past. With the development of test methodologies for Automotive Ethernet, FTZ played an important role in substantiating the feasibility of Automotive Ethernet. As they did so as an independent entity, this added to the credibility of the concept. Many of their results later became part of various OPEN Alliance specifications (e.g., [59] [60] [61]). The UNH-IOL had been included, too, at a very early stage for assessing the feasibility of product development on the basis of the early BroadR-Reach specification. At a later stage UNH-IOL added credibility to the testing of BroadR-Reach/100BASE-T1 components [62].

2 When companies discussed Automotive Ethernet with BMW, they were not only interested in what BMW was doing but also in what everybody else thought. BMW thus perceived a significant market interest and hosted the first "**Ethernet&IP@Automotive Technology Day**" in November 2011; an event, which sold out completely. In alignment with the first Ethernet&IP@Automotive Technology Day, the OPEN Alliance started [63], NXP announced their development of a BroadR-Reach compliant PHY [64] and BMW had completed the internal decisions on the wide introduction of Automotive Ethernet (see Section 3.4.3). In the authors' view, November 2011 represents the turning point. Automotive Ethernet stopped being just an idea of some engineers at BMW and became the future of in-vehicle networking. The Ethernet&IP@Automotive Technology Day allowed for a noncommittal information exchange, with all players present. In 2016 the Technology Day took place for the sixth time. After 2012 was hosted by Continental (with support from Harman) in Regensburg, 2013 was hosted by BOSCH near Stuttgart, 2014 was hosted by GM in Detroit (organized by the IEEE-SA) and 2015 was hosted by JASPAR in Yokohama (organized by Nikkei), the 2016 event was hosted by Renault in Paris [65] (also organized by the IEEE-SA [66]).

Thus the foundation for Automotive Ethernet was laid. Integrating other organizations involved and necessary for Automotive Ethernet (examples are AUTOSAR, AVnu, GENIVI, ISO 17215), setting up future developments at IEEE (especially for higher data rates), and supporting the creation of assurance in the market (open information policy, starting technology days) set additional, reinforcing impulses. In the end, in a growing market a virtuous cycle is achieved between customers, suppliers, and supporting/complementary organizations. For Automotive Ethernet the same cycle is happening. That this is independent from the exact form the end result will have, i.e., what PHY technologies, speed grades, or protocols the industry will use, is one of the strengths of Automotive Ethernet.

The industry acceptance is good. In 2016, three car manufacturers (BMW, JLR, and VW with various brands) publicly stated that they had Automotive Ethernet in series production cars on the road (see, e.g., [67]). At the various events (e.g., the Ethernet&IP@Automotive Technology Days, the Automotive Ethernet Congress and the Hanser Automotive Networks event), additionally, Daimler, GM, Hyundai, PSA Peugeot Citroën, Renault, Toyota, and Volvo Cars have publicly spoken about their use for Automotive Ethernet. At the time of writing GM was chairing the OPEN Alliance in the third year [68] and Volvo Cars provided the Secretary in the second year. And last but not least, next to Broadcom, NXP, Realtek and Marvell had publicly announced 100 and/or 1000BASE-T1 products [69] [70].

Figure 3.9 summarizes the interrelations between the different aspects relevant for the market success of Automotive Ethernet. Last, but not least, there is one element that cannot be structurally captured: the element of chance. The right people with the right skills and ideas have to come across the right potential technical solution in the right innovative environment at the right time. This is what starts technical revolutions.

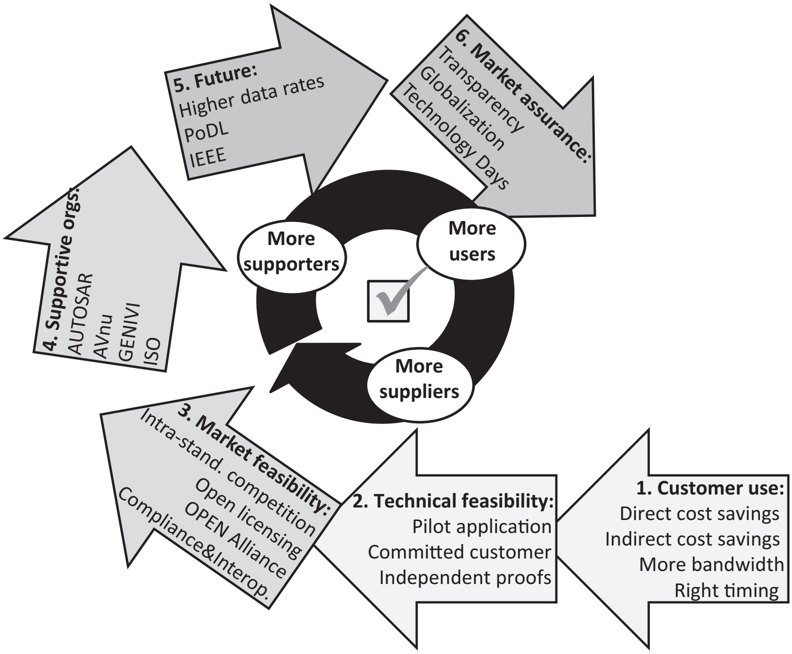

Figure 3.9 Path to Automotive Ethernet.

Notes

1 EMC immunity, sometimes also referred to as Electro Magnetic Susceptibility (EMS), is the ability of a system to function, despite external interference. The EMC immunity tests answer the question of whether a system is stable enough to function correctly in a very bad EMC noise environment (see also Section 4.1).

2 EMC emissions (EME) is the electromagnetic noise generated by the system that via air or cabling can impact the performance of other systems (see also Section 4.1).

3 “At runtime” means that the car is being used for its primary purpose of driving, in contrast to service mode at a garage.

4 Linux is the name for one of the most frequently used operating systems among software developers. It originated during the time that AT&T was engaged in an IP battle with the University of Berkeley over the use of Unix. It was developed as freeware to be POSIX compatible and Unix-like [71].

5 QNX is a commercial operating system that combines Unix principles, real-time and suitability for embedded systems [72].

6 OSEK is an automotive consortium founded in 1993 by players of the German car industry. The most important specifications provided by the consortium describe an embedded operating system, a communications stack, and a network management protocol also suitable for embedded systems. As OSEK was designed to provide standard software architecture for the different ECUs throughout the car [73], many of its principles were reused in the AUTOSAR OS.

7 If a statically charged person or object touches an ElectroStatic Discharge (ESD) sensitive device with a different electrostatic potential, there is a chance that the resultant discharge through the sensitive circuitry will damage the device. The damage can be strong enough

to render the device directly nonfunctional. In a more unfortunate case, the device is simply weakened and failure occurs at a later point in time. With cars consisting of many thousands of parts, it is essential to minimize the risk of such failures and ESD tests are thus an integral part of semiconductor testing. Tests can emulate machines, charged devices, human bodies, and indirect discharges [74] (see also Section 4.1.4).

8 Integrated circuits that have passed the AEC-Q100 qualification program are identified as components suitable for use in the harsh automotive environment. A number of documents provided by the Automotive Electronics Council (AEC) describe in detail the qualification and requalification requirements, test methods, and guidelines [75]. ESD tests are part of AEC-Q100.

9 A year before SOP, development work on the components has normally stopped. The last year focuses on integration and production processes. Thus it was in 2007 that the first introduction of Ethernet in automotive was evaluated strategically and the next steps were being discussed.

10 The same technology has several names. First, it is called “BroadR-Reach,” as this is the name Broadcom, the inventor of the technology, gave the technology and has a trademark on. With BroadR-Reach being facilitated by the OPEN Alliance it is also called OPEN Alliance BroadR-Reach, or OABR, for short. The main characteristic of OABR is that it can be used with Unshielded Twisted Single Pair (UTSP) cabling. At the time of discovery using an Ethernet technology with a single pair worked for just this technology, which is why it is also called “UTSP Ethernet.” We will use this name at the one instance where this is of major importance. However, with the development of an automotive suitable Gbps Ethernet (see Section 4.3.2.1) there is now another PHYs using UTSP Ethernet (and more are being discussed) and “UTSP Ethernet” is no longer unambiguous. At IEEE the UTSP versions have received the suffix “T1.” Therefore 100BASE-T1 is the name BroadR-Reach received as an IEEE standard. As this is the latest name. We will use it whenever possible.

11 Other than the title of the project suggests, the project actually has a strong focus on all protocol layers needed for Automotive Ethernet, while security represented only one of the six work packages defined (see, e.g., [76]). However, also this project had a stronger focus on the protocol than on the PHY layers. More information on the project and its results can be found on [77].

12 The business case was calculated in 2009/2010. At that time pixel links were available for shielded twisted pair cables only and required a separate control channel. Possible also because of the competition from Ethernet pixel links have become significantly cheaper. Not only have the semiconductor prices as such dropped, but newer pixel links developments eliminated the need for a separate control channel, they enable the use of coaxial cabling (which is less expensive than STP) and allow for power transmission with the coaxial (see also Section 2.2.6). A good example of how customers can (occasionally) profit also from intertechnology competition.

13 The definitions of “proprietary,” “open,” and “public” standards are used in this book in accordance with [34]. “Proprietary” is therefore used only for technologies whose IPR is owned by one company that does not license the technology to others under Reasonable and Non-Discriminatory (RAND) terms, but which either licenses very selectively or not at all. In contrast, there can be “open” or “public” technologies. “Public” refers to technologies described in standards developed by Standard Setting Organizations (SSOs) like IEEE or ISO. SSOs follow established rules for IPR, normally requiring that owners of essential patents declare prior to the publication of the standard that they will license their essential patents to interested parties under RAND conditions. “Open” technologies mean that they are at least RAND licensed, regardless of whether the IPR is owned by one or many companies, or whether this is organized in an SSO, a Special Interest Group (SIG) or other consortium, or whether simply the patent holders agree to it. A “public” standard can therefore also be described as “open.” It nevertheless helps for the distinction to refer to “public” standards, if it is a standard published by an SSO.

BroadR-Reach started as a proprietary solution. The OPEN Alliance ensured it became an open technology. With the standardization as IEEE 100BASE-T1 it has become a public standard; this being independent from how many companies own the IP. CAN is another example of a technology that is perceived and accepted as an open and public industry standard, despite the facts that the technology was defined before being made public and that all IPR is owned by BOSCH only, who licenses it under RAND conditions. Both BroadR-Reach and CAN are not "open" following [78]. Here, an open standard requires equal contributions from multiple companies without the dominance of one company. The authors accept this as a different way of looking at it and agree that it is generally (though not always) more motivating for multiple companies to participate in the market, if this is the case. However, for the purposes of this book, the key point is that a technology or standard is not necessarily proprietary, just because one company owns the IPR. It might be open(ed).

14 Like probably all selections of names, the naming of the OPEN Alliance took some time and effort. In the end, One Pair EtherNet (OPEN) reflected best that the BroadR-Reach technology was going to be licensed "openly," i.e., under RAND terms. To the authors, who were involved in the naming, the "One Pair" part of the name was less relevant. It did name a major feature of the BroadR-Reach technology, but in the end OPEN's purpose was and is facilitating Ethernet-based communication in automotive, independent from whether one or multiple pairs or even other medias are used.

15 The public announcements on this event are not 100% correct. The reader has to trust that the authors as founders and first chair of the OPEN Alliance know better.

16 Network effects refer to products whose use increases with distribution [78] [79]. Communication technologies per definition require communication partners; the more there are, the higher the use of the technology for all. Inside vehicles it is a little different because after all it is the car manufacturer who decides on (almost) all communication interfaces inside the car. Nevertheless, a large distribution among other car manufacturers has more advantages than just better economies of scale. It leads to a better educated workforce, better suited tooling, better infrastructure in terms of independent test houses, more reliability from the Tier 1s, etc. The network effects are thus indirect.

17 The name "Power over Ethernet" (PoE) is tied to the IEEE 802.3af standards and their successors. It is also referred to as "clause 33" into which it was incorporated in the standards revision IEEE 802.3–2005. Clause 33 implies a specific method requiring two twisted pairs of cabling. A standard that was going to discuss the transmission of power over one pair only therefore needed a different name. Instead of "1 Pair Power over Ethernet" the activity is thus called "1 Pair Power over Data Line." In contrast, the IEEE 802.3 activity that investigates the transmission of power over data lines with four pairs is called "4 Pair Power over Ethernet" [72].

References

[1] A. L. Russell, "OSI: The Internet That Wasn't," 29 July 2013. [Online]. Available: http://spectrum.ieee.org/computing/networks/osi-the-internet-that-wasnt. [Accessed 22 August 2013].

[2] U. v. Burg and M. Kenny, "Sponsors, Communities, and Standards: Ethernet vs. Token Ring in the Local Area Networking Business," *Industry and Innovation*, vol. 10, no. 4, pp. 351–375, December 2003.

[3] R. M. Metcalfe, "The History of Ethernet," 2006. [Online]. Available: www.youtube.com/watch?v=g5MezxMcRmk. [Accessed 26 Febrary 2017.

[4] Wikipedia, "On-board diagnostics," 1 September 2013. [Online]. Available: http://en.wikipedia.org/wiki/On_Board_Diagnostics. [Accessed 2 September 2013].

[5] K. Matheus, "OPEN Alliance – Stepping Stone to Standardized Automotive Ethernet," in *2nd Ethernet&IP@Automotive Technology Day*, Regensburg, 2012.

[6] BMW, "Ethernet Diagnose Stecker," April 2007. [Online]. Available: http://bmwtools.info/uploads/enet_doku.pdf. [Accessed 29 September 2013].

[7] ISO, "ISO 14229–1:2013–03 (E) Road vehicles – Unified diagnostic services (UDS) – Part 1: Specification and requirements," ISO, Geneva, 2013.

[8] Free Software Foundation, "lwIP – A Lightweight IP stack – Summary," 2013. [Online]. Available: http://savannah.nongnu.org/projects/lwip/. [Accessed 28 September 2013].

[9] UNECE (United Nations Economic Commission for Europe), "gtr No. 4," 1998. [Online]. Available: www.unece.org/fileadmin/DAM/trans/main/wp29/wp29wgs/wp29gen/wp29registry/ECE-TRANS-2005–124–04e.pdf. [Accessed 3 August 2016].

[10] M. Johanson, P. Dahle, and A. Söderberg, "Remote Vehicle Diagnostics over the Internet using the DoIP Protocol," in *Proceedings of the Sixth International Conference on Systems and Networks Communications*, Barcelona, October 2011.

[11] ISO, "ISO 13400–1:2011 – Road vehicles – Diagnostic Communication over Internet Protocol (DoIP)," ISO, Geneva, 2011.

[12] G. Mascolino, "Ethernet – der Standard. Jetzt auch im Automobil," in *Automotive Kongress*, Ludwigsburg, 2012.

[13] R. Bruckmeier, "Ethernet for Automotive Applications," in *Freescale Technology Forum*, Orlando, 2010.

[14] M. Jones, "Ethernet im Automobil," *Automobil Elektronik*, pp. 40–42, 01 2011.

[15] IEEE 802.3, "Reduced Twisted Pair Gigabit Ethernet Call for Interest," March 2012. [Online]. Available: www.ieee802.org/3/RTPGE/public/mar12/CFI_01_0312.pdf. [Accessed 15 March 2012].

[16] InvestmentMine, "Historical Copper Prices and Price Chart," 26 September 2013. [Online]. Available: www.infomine.com/investment/metal-prices/copper/all/. [Accessed 3 August 2016].

[17] Heise Online, "BMW erforscht Bordnetz mit Internet-Protokoll," 27 October 2007. [Online]. Available: www.heise.de/autos/artikel/BMW-erforscht-Bordnetz-mit-Internet-Protokoll-466769.html. [Accessed 23 July 2016].

[18] IEEE 802.3, "Residential Ethernet, IEEE 802.3 Call for Interest," July 2004. [Online]. Available: http://grouper.ieee.org/groups/802/3/re_study/public/200407/cfi_0704_1.pdf. [Accessed 12 October 2013].

[19] J. Hillebrand, M. Rahmani, R. Bogenberger, and E. Steinbach, "Coexistence of Time-Triggered and Event-Triggered Traffic in Switched Full-Duplex Ethernet Networks," in *Industrial Embedded Systems, 2007. SIES '07, International Symposium on Digital Object Identifier*, Lisbon, 2007.

[20] M. Rahmani, J. Hillebrand, W. Hintermaier, R. Bogenberger, and E. Steinbach, "A Novel Network Architecture for in-Vehicle Audio and Video Communication," in *2nd IEEE/IFIP International Workshop on Broadband Convergence Networks, 2007. BcN '07*, Munich, 2007.

[21] M. Rahmani, R. Steffen, K. Tappayuthpijarn, E. Steinbach, and G. Giordano, "Performance Analysis of Different Network Topologies for in-Vehicle Audio and Video Communication," in *4th International Telecommunication Networking Workshop on QoS in Multiservice IP Networks, 2008*, Venice, 2008.

[22] M. Rahmani, A. Pettiti, E. Biersack, E. Steinbach, and J. Hillebrand, "A Comparative Study of Network Transport Protocols for in-Vehicle Media Streaming," in *IEEE International Conference on Multimedia and Expo, 2008*, Hanover, 2008.

[23] M. Rahmani, K. Tappayuthpijarn, B. Krebs, E. Steinbach, and R. Bogenberger, "Traffic Shaping for Resource-Efficient in-Vehicle Communication," *IEEE Transactions on Industrial Informatics*, pp. 414–428, 15 May 2009.

[24] C. Salzmann, Modified from nonpublic document, Munich, 2009.

[25] T. Grünweg, "Eine Industrie kommt auf Speed," 10 February 2013. [Online]. Available: www.spiegel.de/auto/aktuell/warum-lange-entwicklungszyklen-fuer-autohersteller-zum-problem-werden-a-881990.html. [Accessed 15 February 2013].

[26] K. Matheus, "Ethernet-basierte Kommunikation: Der skalierbare Vernetzungsbaukasten für die Fahrzeugentwicklung," *Elektonik Automotive*, January 2013.

[27] M. Rahmani, E. Steinbach, W. Hintermaier, A. Laika, and H. Endt, "A Novel Network Design for Future IP-Based Driver Assistance Camera Systems," in *International Conference on Networking, Sensing and Control, 2009, ICNSC '09*, Okayama, 2009.

[28] W. Hintermaier and E. Steinbach, "A System Architecture for IP-Camera Based Driver Assistance Applications," in *2010 IEEE Intelligent Vehicles Symposium (IV)*, San Diego, 2010.

[29] T. Hase, W. Hintermaier, A. Frey, T. Strobel, U. Baumgarten and E. Steinbach, "Influence of Image/Video Compression on Night Vision Based Pedestrian Detection in an Automotive Application," in *2011 IEEE 73rd Vehicular Technology Conference (VTC Spring)*, Yokohama, 2011.

[30] T. Königseder and S. Singer, "Development Partnership BMW-Freescale Enables Ethernet for Automotive Applications," in *Freescale Technology Forum*, Austin, 2011.

[31] K. Balszuweit, "Einführung Ethernet&IP als applikativer Fahrzeugbus," in *Steinbeis Symposium*, Fellbach, 2013.

[32] K. Matheus, "Structural Support for Developing Automotive Ethernet," in *3rd Ethernet&IP@Automotive Technology Day*, Leinfeld-Echterdingen, 2013.

[33] Center of Automotive Management, "Automotive INNOVATIONS Award 2012," 2 May 2013. [Online]. Available: www.auto-institut.de. [Accessed 9 May 2013, no longer available].

[34] F. Borowitz and E. Scherm, "Standardisierungsstrategien, eine erweiterte Betrachtung des Wettbewerbs auf Netzeffektärkten," Fernuniversität Hagen, Hagen, 1999.

[35] H. L. Gabel, *Produktstandardisierung als Wettbewerbsstrategie*, London: McGraw-Hill, 1993.

[36] OPEN Alliance, "Non-Profit Alliance Reached a New Milestone: Over 300 Members," 22 May 2016. [Online]. Available: http://opensig.org/news/press-releases/. [Accessed 3 August 2016].

[37] N. A. Wienckowski, "Enabling 1000BASE-T1 for Automotive Applications," in *IEEE-SA (6th) Ethernet&IP@Automotive Technology Day*, Paris, 2016.

[38] C. Hill, "Establishing a Standard: Competitive Strategy and Technological Standards in Winner-Take-All Industries," *Academy of Management Executive*, vol. 11, no. 2, pp. 7–25, 1997.

[39] B. D. Abramson, "The Patent Ambush: Misuse or Caveat Emptor?," 5 January 2011. [Online]. Available: www.ipmall.info/sites/default/files/hosted_resources/IDEA/idea-vol51-no1-abramson.pdf. [Accessed 3 August 2016].

[40] IEEE 802.3, “Approved Minutes IEEE 802.3 Ethernet Working Group PLENARY Grand Hyatt, San Antonio, TX USA,” 14–18 November 2012. [Online]. Available: www.ieee802.org/3/minutes/nov12/minutes_1112.pdf. [Accessed 31 August 2016].

[41] IEEE 802.3, “Power over Data Lines Call for Interest,” 16 July 2013. [Online]. Available: www.ieee802.org/3/cfi/0713_1/CFI_01_0713.pdf. [Accessed 31 July 2013].

[42] IEEE 802.3, “IEEE 802.3 Ethernet Working Group,” IEEE 802.3, 2016. [Online]. Available: www.ieee802.org/3/. [Accessed 4 July 2016].

[43] L. Winkel, M. McCarthy, D. Brandt, and K. Matheus, “10Mb/s Single Twisted Pair Ethernet Call for Interest,” 26 July 2016. [Online]. Available: www.ieee802.org/3/cfi/0716_1/CFI_01_0716.pdf.

[44] F. Leitner, “AUTOSAR AUTomotive Open System ARchitecture,” 2007. [Online]. Available: www.inf.uni-konstanz.de/soft/teaching/ws07/autose/leitner-autosar.pdf. [Accessed 9 October 2013, no longer available].

[45] AUTOSAR, “Requirements on Ethernet Support in AUTOSAR, Release 4.0, Revision 1,” 7 December 2009. [Online]. Available: www.autosar.de/download/R4.0/AUTOSAR_SRS_Ethernet.pdf. [Accessed 9 October 2013, no longer available].

[46] AUTOSAR, “Specification of Ethernet Interface, Release 4.1., Revision 3,” 31 March 2014. [Online]. Available: www.autosar.org/fileadmin/files/releases/4–1/software-architecture/communication-stack/standard/AUTOSAR_SWS_EthernetInterface.pdf. [Accessed 3 August 2016].

[47] AUTOSAR, “Specification of Ethernet Interface, Release 4.2.2,” 2015. [Online]. Available: www.autosar.org/fileadmin/files/releases/4–2/software-architecture/communication-stack/standard/AUTOSAR_SWS_EthernetInterface.pdf. [Accessed 3 August 2016].

[48] Business Wire, “AVnu Alliance Launches to Advance Quality of Experience for Networked Audio and Video,” 25 August 2009. [Online]. Available: www.businesswire.com/news/google/20090825005929/en. [Accessed 8 October 2013].

[49] AVnu Alliance, “Homepage of the AVnu Alliance,” AVnu Alliance, 2013. [Online]. Available: http://avnu.org/. [Accessed 8 October 2013].

[50] AVnu Alliance, “Industrial Control & Monitoring,” 2016. [Online]. Available: http://avnu.org/industrial/. [Accessed 23 July 2016].

[51] M. Jochim and J. Specht, “AAA2C – Automotive Requirements for a Flexible Control Traffic Class,” 14 July 2014. [Online]. Available: www.ieee802.org/1/files/public/docs2013/new-tsn-jochim-aaa2c-requirements-for-control-traffic-0713-v01.pdf. [Accessed 7 July 2016].

[52] AVnu Alliance, “AVnu Knowledge Base, AVnu Automotive Advisory Council (AAA2C) Materials,” AVnu Alliance, 2016. [Online]. Available: http://avnu.org/whitepapers/. [Accessed 7 July 2016].

[53] G. Bechtel, B. Gale, M. Kicherer, and D. Olsen, “Automotive Ethernet AVB Functional and Interoperability Specification Revision 1.4,” 12 May 2015. [Online]. Available: http://avnu.org/wp-content/uploads/2014/05/AutoCDSFunctionalSpec-1_4-public_with_legal_notices.pdf. [Accessed 7 July 2016].

[54] ISO, “ISO/DIS 17215 Road Vehicles – Video Communication Interface for Cameras (VCIC) Parts 1 – 4,” ISO, Geneva, 2013.

[55] O. Sugihara and S. Takahashi, “Standardization Activities on Gigabit Ethernet Operation over Plastic Optical Fiber,” in *5th Ethernet&IP@Automotive Technology Day*, Yokohama, 2015.

[56] F. Horikoshi, "NWIP on Road Vehicles – In-Vehicle Gigabit Ethernet System – Part 1: General Requirements of Gigabit Ethernet System and Physical and Data-Link Layer Requirements of Optical Gigabit Ethernet System," in *ISO*, Geneva, 2015.

[57] F. Horikoshi, "NWIP on Road Vehicles – In-Vehicle Gigabit Ethernet System – Part 3: General Requirements and Test Methods of Optical Gigabit Ethernet Components," in *ISO*, Geneva, 2015.

[58] B. Wuelfing, "CeBIT 2009: BMW and Partners Found GENIVI Open Source Platform," 3 March 2009. [Online]. Available: www.linuxpromagazine.com/Online/News/CeBIT-2009-BMW-and-Partners-Found-GENIVI-Open-Source-Platform. [Accessed 10 October 2013].

[59] B. Körber, "EMC Test Specification for BroadR-Reach™ Transceivers, Version 1.1," OPEN Alliance, 2013.

[60] B. Körber, "EMC Test Specification for BroadR-Reach™ Common Mode Chokes, Version 1.1," OPEN Alliance, 2013.

[61] B. Körber, "BroadR-Reach Physical Layer, Definitions for Communication Channel," OPEN Alliance, 2013.

[62] Business Wire, "OPEN Alliance and UNH-IOL Rev up Evolution of Connected Car," 20 August 2012. [Online]. Available: www.businesswire.com/news/home/20120820005361/en/OPEN-Alliance-UNH-IOL-Rev-Evolution-Connected-Car. [Accessed 11 October 2013].

[63] PR Newswire, "Broadcom, NXP, Freescale, and Harman Form OPEN Alliance Special Interest Group," 9 November 2011. [Online]. Available: www.prnewswire.com/news-releases/broadcom-nxp-freescale-and-harman-form-open-alliance-special-interest-group-133514928.html. [Accessed 9 November 2011].

[64] marketwired, "NXP Develops Automotive Ethernet Transceivers for In-Vehicle Networks," 9 November 2011. [Online]. Available: www.marketwired.com/press-release/nxp-develops-automotive-ethernet-transceivers-for-in-vehicle-networks-nasdaq-nxpi-1584249.htm. [Accessed 9 November 2011].

[65] IEEE-SA, "IEEE Standards Association (IEEE-SA) Ethernet&IP@Automotive Technology Day," IEEE-SA, 2016. [Online]. Available: https://standards.ieee.org/events/automotive/.

[66] S. Yu, "IEEE Standards Association (IEEE-SA) Exhibiting at Ethernet&IP@Automotive Technology Day in 2013, Hosting Event Starting in 2014," 11 September 2013. [Online]. Available: http://standards.ieee.org/news/2013/ethernet-technology-day.html. [Accessed 8 October 2013].

[67] OPEN Alliance, "Automotive Ethernet Hits the Road in Wide Range of New Vehicles," 14 October 2015. [Online]. Available: http://opensig.org/news/press-releases/. [Accessed 3 August 2016].

[68] OPEN Alliance, "World's Leading Auto Makers, Tier Ones and Tech Companies Partner to Drive Wide Scale Adoption of Automotive Ethernet," 9 June 2014. [Online]. Available: http://opensig.org/news/press-releases/. [Accessed 3 August 2016].

[69] Marvell, "Marvell Unveils Industry's First 1000BASE-T1 Automotive Ethernet PHY Transceiver," October 2015. [Online]. Available: www.marvell.com/company/news/pressDetail.do?releaseID=7256. [Accessed 23 July 2016].

[70] OPEN Alliance, "Automotive OPEN Alliance Demonstrates Multi-Vendor Interoperability," 25 October 2015. [Online]. Available: http://opensig.org/news/press-releases/. [Accessed 3 August 2016].

[71] Wikipedia, "Geschichte von Linux," 14 July 2016. [Online]. Available: http://de.wikipedia.org/wiki/Geschichte_von_Linux. [Accessed 3 August 2016].
[72] Wikipedia, "QNX," 21 July 2016. [Online]. Available: http://en.wikipedia.org/wiki/QNX. [Accessed 3 August 2016].
[73] Wikipedia, "OSEK," 18 July 2016. [Online]. Available: http://en.wikipedia.org/wiki/OSEK. [Accessed 3 August 2016].
[74] ON Semiconductors, "In Vehicle Networking ESD Performance," [Online]. Available: www.onsemi.com/pub_link/Collateral/TND391-D.PDF. [Accessed 29 September 2013, no longer available].
[75] Automotive Electronics Council, "AEC Documents," 17 July 2012. [Online]. Available: www.aecouncil.com/AECDocuments.html. [Accessed 29 September 2013].
[76] M. Glaß, D. Herrscher, H. Meier, M. Piastowski, and P. Schoo, "SEIS – Sicherheit in Eingebetteten IP-Basierten Systemen," *ATZelektronik*, vol. 5, no. 1, pp. 50–55, 2010.
[77] Universität Erlangen, Lehrtstuhl für Informatik 12, "SEIS – Sicherheit in Eingebetteten IP-Basierten Systemen," 6 June 2013. [Online]. Available: www12.informatik.uni-erlangen.de/research/seis/. [Accessed 23 July 2016].
[78] M. Katz and C. Shapiro, "System Competition and Network Effects," *Journal of Economic Perspectives*, vol. 8, no. 2, pp. 93–115, 1994.
[79] J. Farrel and G. Saloner, "Standardization, Compatibility, and Innovation," *RAND Journal of Economics*, vol. 16, no. 1, pp. 70–83, 1985.
[80] S. Yu, "IEEE Forms 4-Pair Power over Ethernet (PoE) Study Group," 1 April 2013. [Online]. Available: http://standards.ieee.org/news/2013/4pair_poe.html. [Accessed 7 October 2013].

4 The Physical Transmission of Automotive Ethernet

To understand the physical transmission in in-vehicle networks two aspects are of importance: The actual automotive environment in which the communication happens and how the properties of the physical layer (PHY) technology ensure that the PHY meets the requirements in this environment. It is therefore common to first define the environment and then to develop the PHY in this environment. One of the key challenges in the automotive environment is meeting the stringent ElectroMagnCompatibility (EMC) requirements. This chapter therefore starts in Section 4.1 with explaining EMC in the automotive context. It then describes in Section 4.2 the transmission channel, which has to meet – among other aspects – the automotive EMC requirements, before coming to the main part of this chapter: The different Automotive Ethernet PHY technologies are described in Section 4.3.

However, there is more to the physical transmission. A very important aspect is the power supply and power consumption. Section 4.4 discusses methods to transmit power with the Ethernet data (Power over DataLine, PoDL) and methods to save power like Energy-Efficient Ethernet (EEE) and wake-up, which all impact the development of the physical transmission system. Last but not least, Section 4.5 addresses the challenges in respect to the quality requirements active and passive components have to meet in order to be used in cars. Both topics impact the PHY design, but go beyond and/or are independent of the actual PHY specification used and have therefore been placed following the EMC, channel, and PHY technology sessions at the end of this chapter. Figure 4.1 gives an overview on the different sections and how they relate.

4.1 ElectroMagnetic Compatibility (EMC)

If a device is electromagnetically compatible this means that it functions in its intended surroundings without being impaired by electromagnetic emissions of other devices in the same physical location while not disturbing the performance of other devices by its own emissions [1]. Both, the ElectroMagnetic Immunity (EMI)[1] against interference from others as well as the own ElectroMagnetic Emissions (EME) are integral parts of the EMC performance of a device [2].

EMC has a long history, in which the automobile actually accelerated the respective legislation. The first ever law on the topic was passed in 1892 in Germany in the context of the upcoming telegraph and telephone business [2]. It had become evident at an early

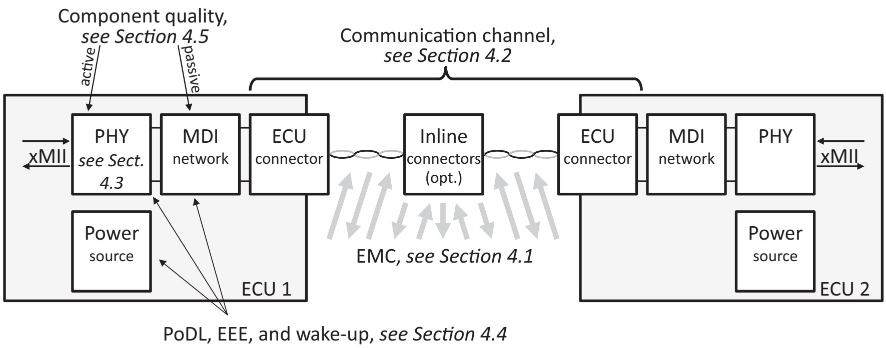

Figure 4.1 Elements of physical transmission discussed and their interrelation.

stage that physically close cables can interfere with each other's transmissions. This interference was especially painful in case of telegraph and telephone lines. The law thus dealt with the impact such interference had on respective devices and installations and how to handle it.

However, the EMC topic received a push in Germany on 22 December 1920 with a life radio transmission of a Christmas concert southeast of Berlin. The German chancellor of the time, was invited to a nearby location in order to be charmed by the latest technical achievements, but instead was angered by the crackling every passing car induced in the speakers. Countermeasures had to be taken and – what was only later called – EMC was by 1927 the reason for the first German law on the use and installation of high frequency radio transmitters. The law included limit lines and a clearance process, which were, with adaptations, valid in Germany until 1995 [2].

The international community saw similar developments. In 1933 the Comité International Spécial des Perturbations Radioélectriques (CISPR) was founded in Paris in order to develop guidelines on a European level. In the US, the American National Standards Institute (ANSI) and Federal Communications Commission (FCC) also produced respective rules for the US [3]. However, the need to regulate EMC on an even much broader scale arose with the invention and spread of the transistor. In 1973 the International Electrotechnical Commission (IEC) created a special technical committee with the purpose to handle EMC topics [2]. Today, with electronics having penetrated into every part of everyday life, EMC is more important than ever.

This book discusses three perspectives on EMC:

1 The coupling mechanisms, i.e., how the electric and electronic activity of one unit can actually affect the performance of another.
2 The standards addressing EMC.
3 The test methods to evaluate the EMC behavior.

All three approaches are briefly described in order to generate the necessary understanding of the requirements for Automotive Ethernet in general and of the actual results achieved with 100BASE-T1/BroadR-Reach Ethernet in an automotive environment specifically (see Section 4.1.3.1). Next to emissions and immunity, the

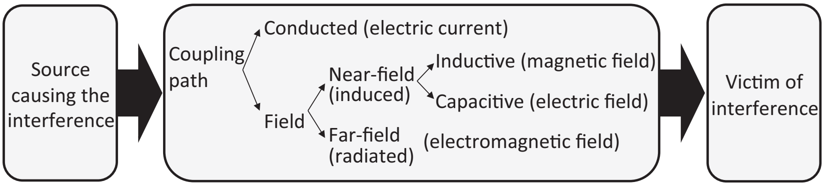

Figure 4.2 EMI model with source, coupling (types), and victim (sink).

ElectroStatic Discharge (ESD) is also important for the quality and lifetime of an electronic device. Even though ESD is not strictly part of EMC, it is part of the qualification tests that need to be performed and thus included in this subsection (see Section 4.1.4).

4.1.1 Coupling Mechanisms of Electromagnetic Interference

In principle, every electronic device can at the same time be the cause as well as the victim of electromagnetic interference. It thus makes sense to select one device as the victim while identifying possible sources and coupling mechanisms that cause the interference (see Figure 4.2 or, e.g., [4]). The coupling mechanisms can be grouped into conducted coupling and coupling caused by a field. The field coupling can be far-field or near-field. In the latter case source and sink are less than one-sixth of a wave length apart [5] and the coupling can be inductive from a magnetic field and capacitive from an electric field [6]. All four coupling paths can coexist and disturb one device at the same time. In order to countermeasure the interference, the correct identification of coupling paths and sources is important.

- In case of **conducted coupling** unintended signal energy leaves a unit via its cables. An example is High Frequency (HF) energy coupling into and leaving a device via the power supply cable, where it is not meant to be and from where it can cause interference to other devices directly from the power supply [6]. This type of interference is not inhibited by inline connectors. Often insufficient or defect ground connections, causing so-called ground loops, enhance this interference. If two units theoretically use the same ground but one has a significantly longer distance to it, or one ground connector is simply not functioning well, an interference that would otherwise simply be led to ground, might find another path with lower impedance. Conducted coupling can thus also couple galvanically. A common mode signal coupled onto a wire pair causes currents flowing in the same direction on both wires. A differential signal causes opposing effects on both wires. Filtering and proper ground measures, which need to take the complete car design into account, combat this type of coupling.
- In case of **near-field coupling**, interference is induced into a victim by a changing electric or magnetic field that is at a closer distance than one-sixth of the wave length. The interference consequently increases with faster changes in the field, higher frequencies, and shorter distance [6]. Figure 4.3 shows the principle functioning of capacitive and inductive coupling in one schematic. For **capacitive coupling**, i.e., coupling from an electric field, the voltage of the interference source V_S causes an

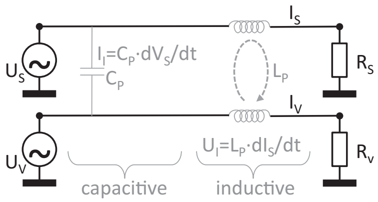

Figure 4.3 Near-field coupling via a parasitic capacitor or inductor [6].

electric field across the gap between its own wire and the wire of an adjacent victim (V) system. The induced/interfering current I_I depends on the change of the voltage U_I and the parasitic capacitor C_P the units share. Typical sources for electric, i.e., capacitive coupling are high-voltage power lines, ignition systems or transceivers [6]. They represent very different technologies, but all have a high impedance in common. For **inductive coupling**, i.e., coupling from a magnetic field, the current in the wire from the source (S) system induces a magnetic field and thus a voltage into the victim system that depends on the parasitic inductor L_P. Typical examples for inductive interferers are highway control transmitters, wireless stations, and radio frequency transmitters. As said, it is circuits with high impedance that are more likely to couple capacitively. Circuits with low impedance are more likely to cause interference from inductive coupling [6].

In communication systems, including Ethernet, one type of near-field electromagnetic interference is referred to as crosstalk. In case of crosstalk a differential signal couples into another differential signal. Near-End Cross Talk (NEXT) and Far-End Cross Talk (FEXT) cause interference induced by an electric or magnetic field from wires of the same system, while Alien NEXT (ANEXT) and Alien FEXT (AFEXT) are from neighboring wires of another system. Using shielded cables increases the immunity of cables against (A)FEXT and (A)NEXT as well as against common mode interference. Additionally it reduces the emissions, provided the shield has a low impedance ground connection and that the shield itself does not carry interference. The complete ECU design and situation in the vehicle needs to be taken into account, when using a shielded cable as EMC protection. However, because of the high costs of shielded solutions, the use of unshielded solutions is preferred in the automotive industry. Twisting the wires helps to improve the performance in case of differential transmissions, as the two wires of the twisted pair are subject to the same electromagnetic field. The coupling therefore is the same on both wires, which means that the differential signal is not affected as much, because the common mode interference is eliminated when the differential signal is combined at the receiver. The more symmetric, i.e., the more balanced a twisted cable pair is, the better. Furthermore, the EMC crosstalk performance can be improved, if the distance between two potentially conflicting cables or wires is increased. In Automotive Ethernet using a jacket can cause the essential difference (see, e.g., Section 4.3.2.1 for RTPGE/1000BASE-T1).

Note that transient disturbances from generators typical in automotive are not that critical in case of Automotive Ethernet. Because the transmit spectrum of Automotive

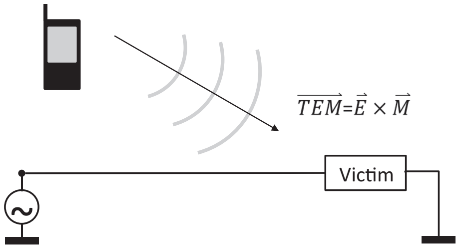

Figure 4.4 Radiated far-field coupling (also known as Radio Frequency Interference (RFI)) [7].

Ethernet is comparably high, the low frequency transients can be suppressed with standard filters. For traditional in-vehicle networking systems operating in a lower frequency range the interference from these transient disturbances is a more serious concern.

- When the distance between the interference source and the interfered sink increases, only radiated **far-field coupling** can be the cause of electromagnetic interference. Mobile devices like mobile phones that use a transmitter are per se a potential source of such interference as it is their purpose to transmit electromagnetic energy through space in order to communicate. The phones radiate so-called Transversal ElectroMagnetic (TEM) waves that consist of an electric and a magnetic component (see also Figure 4.4). Any circuit that contains antenna-like elements for the right frequency will receive some of the energy transmitted and thus experience interference [6]. Cell phones used in a car may produce noise on both signal and power lines. However, the coupled energy is normally significantly smaller than the required automotive limits that are tested, e.g., with the BCI test (see also Section 4.1.3).

4.1.2 Standards for EMC

The topic of EMC is complex and requires a significant amount of experience and references. The existing standards are thus often based on prior versions and reuse or reference the experience that has been collected over decades. In addition to the relevant ISO standards that are listed in Table 4.1, various earlier and various national norms, or norms that are used with national preference, exist, e.g., from the VDE, CISPR, or IEC (see also Section 4.1.3). However, with the increasing globalization, the international applicability and up-to-datedness makes the ISO standards the most comprehensive documents. Table 4.1 serves as an introduction and orientation.

4.1.3 Measuring EMC

Cars are the skillful combination of thousands of parts from different sources. In order to provide optimal quality, tests and validations are performed on all levels of the car development and production. Proving the EMC performance is an integral part of this. Respective tests are performed on semiconductor, ECU, and vehicle level.

Table 4.1 Overview of ISO EMC standards

Standard	Content
ISO 7637–1	Road vehicles: Electrical disturbances from conduction and coupling Part 1: Definitions and general considerations Replaces DIN 40839
ISO 7637–2	Road vehicles: Electrical disturbances from conduction and coupling Part 2: Electrical transient conduction along supply lines
ISO 7637–3	Road vehicles: Electrical disturbances from conduction and coupling Part 3: Electrical transient transmission by capacitive and inductive coupling via lines other than supply lines
ISO 10605	Road vehicles: Test methods for electrical disturbances from electrostatic discharge
ISO 11452–1	Road vehicles: Component test methods for electrical disturbances from narrowband radiated electromagnetic energy Part 1: General principle and terminology
ISO 11452–2	Road Vehicles: Component test methods for electrical disturbances from narrowband radiated electromagnetic energy Part 2: Absorber-lined shielded enclosure
ISO 11452–3	Road Vehicles: Component test methods for electrical disturbances from narrowband radiated electromagnetic energy Part 3: TEM-Cell
ISO 11452–4	Road Vehicles: Component test methods for electrical disturbances from narrowband radiated electromagnetic energy Part 4: Harness excitation methods (Bulk Current Injection (BCI))
ISO 11452–5	Road Vehicles: Component test methods for electrical disturbances from narrowband radiated electromagnetic energy Part 5: Stripline
ISO 11452–8	Road Vehicles: Component test methods for electrical disturbances from narrowband radiated electromagnetic energy Part 8: Immunity to magnetic fields

To perform tests on semiconductor level is comparably new for car manufacturers, but especially crucial when introducing a new in-vehicle networking technology like Automotive Ethernet. After all, it is the Tier 1 suppliers that are responsible for the correct functioning of the ECUs they deliver, but the car manufacturer who decides on and is responsible for a functioning in-vehicle network (see also Section 2.3.2). The earlier in the development process potential error sources are being detected, the less likely are malfunction and the easier to handle, at a later point in time. To solve potential EMC issues on semiconductor level is about as early as it is possible to find errors in a system.

The most conclusive results on semiconductor immunity are achieved with the Direct Power Injection (DPI) [8]. The DPI test is defined in the IEC 62132–4 standard. When deploying the DPI test with Automotive Ethernet, the developer has to pay attention to the fact that the DPI test impacts the return loss performance. In case of in-vehicle networking technologies with shared medium – like CAN, LIN, or FlexRay – this has no consequence. The link is always used for one transmission direction only. However, in case of full-duplex Automotive Ethernet, the test setup of the DPI test can make the link perform worse than it would be without the test setup. In order to avoid any test artifacts, the coupling termination in the DPI test setup thus has to be matched carefully in order to ensure that the impedance of the transmission link does not change [9].

An important (conducted) emissions test on semiconductor level is the "150 Ohm direct coupling method," also described in IEC 62132–4. 150 Ohms is the typical impedance of an in-vehicle wiring harness. For the tests, thus a decoupling network with 150 Ohm impedance is used to measure the frequency spectrum of the output voltage at certain IC pins or IC pin groups [10]. With test boards similar to the ones used for the DPI tests and special measurement receivers the emissions can be assessed. If a semiconductor passes the DPI and 150 Ohm tests, this is a first assurance that its design will show a good EMC performance also when being integrated into an ECU and later a car. In consequence car manufacturers might request the positive test results before recommending the semiconductor to be used by their Tier 1s.

The tests on ECU level endeavor to emulate the automotive environment and consist of measurements performed on the ECU in the laboratory. For the tests, the communication interfaces and their communication partners are modeled in order to achieve useful results. In the laboratory, ElectroMagnetic Emissions (EME) as well as Immunity (EMI) are measured extensively. The ISO tests Bulk Current Injection (BCI) and stripline measurements are common. Also, TEM-cell as well as antenna tests in the absorber-lined chamber are typical. The current injection during the BCI test simulates an external EMC noise being coupled into the wiring harness. For the stripline and TEM-cell measurements special antennas are used to simulate specific interference with repeatable conditions.

Finally, the technology is integrated and measured inside the car. This is extremely important, as the assumptions made in the laboratory are not fully applicable inside the car. Additionally, not 100% of the in-vehicle effects can be modeled; sometimes simply because they are not known in advance. Examples are the effects of the ground connection and the electric fields that depend on the exact body form. Simulations and laboratory tests are continuously being improved, but because of the complexity of the coupling effects in the real product, simulation results only give an indication. Before a technology has not been qualified with in-vehicle measurements, it is not ready for production. Table 4.2 gives an overview on the hierarchy of important EMC measurement methods in automotive; without making claims to be complete.

4.1.3.1 EMC Results for 100BASE-T1/OABR

This section shows examples of typical EMC test results for 100BASE-T1/OABR Ethernet in relation to CAN. CAN has been selected for comparison, because it is a well-established technology in the industry whose EMC performance – in contrast to that of 100BASE-T1/OABR – is not questioned. The measurement results show a sequence, from 150 Ohm emission measurements using a test board (Figure 4.5), via DPI tests using a test board (Figure 4.6), to stripline emission tests with a reference ECU (Figure 4.7). For both CAN and 100BASE-T1/OABR, better as well as worse

Table 4.2 Example hierarchy of EMC measurement methods in automotive

	Semiconductor		ECU		Vehicle
Immunity	• Direct Power Injection (DPI), IEC 62132–4	→	• Bulk Current Injection (BCI), ISO 11452–4 • Transversal ElectroMagnetic (TEM) cell, ISO 11452–3 • Antenna measurements in absorber-lined chambers, ISO11451–2	→	• Antenna measurements in absorber-lined chambers, ISO 11451–2 (orig. CISPR25) • Stripline, ISO11452–5 with OEM adaptations for large cars
Emissions	• 150 Ohm method, IEC 62132–4	→	• Stripline, ISO 11452–5 • Antenna measurements in absorber-lined chambers, ISO 11451–2	→	• Measurements with vehicle on-board antennas, ISO 11451–2 (orig. CISPR12, EN55025) • Antenna measurements in absorber-lined chambers ISO 11451–2 (orig. CISPR25)

Note: The tests generally vary from car manufacturer to car manufacturer.

results can be achieved depending on the actual implementations. The examples shown give an orientation and present the limit lines that need to be met.

Conclusion: The 100BASE-T1/OABR Ethernet tests on board level meet the automotive requirements. Also when comparing the results with CAN, they are good. Of course, the frequency behavior of the EME is different as a different communication method is used. While CAN works in a range of 200 kHz to a few MHz, 100BASE-T1/OABR Ethernet is using a bandwidth up to 33⅓ MHz. The peak that can be seen

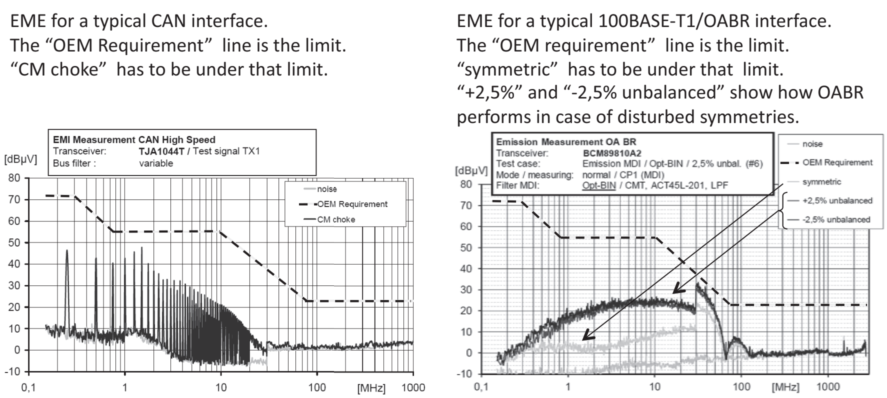

Figure 4.5 Example results for 150 Ohm EME tests according to IEC 61967–4.

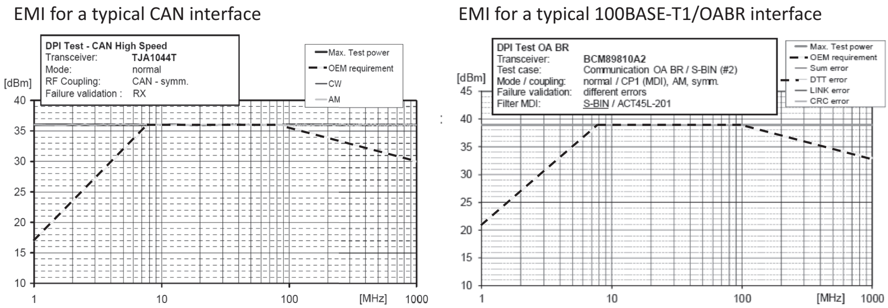

Figure 4.6 Example results for DPI tests according to IEC 61967–4. The level of noise stimulating the system has to be over the "OEM Requirement." If the communication in the system is working under such noisy conditions, the test is passed. Note that all measured curves in both pictures are superposed; the limit line in the OABR picture is somewhat stricter because it represent a recently harmonized limit line.

at 30 MHz in the OABR curves is an artifact of the measuring equipment, which changes the resolution just at this frequency.

Conclusion: 100BASE-T1/OABR passes the immunity requirements of many car manufacturers, just like CAN does.

Conclusion: 100BASE-T1/OABR Ethernet is working under real automotive conditions and the limits of the EME can be passed.[2] The results of the stripline emission test with the reference ECU fits very well to the results achieved with the test board using the 150 Ohm method (see Figure 4.5). As it is therefore seen as redundant, the book thus does not show additional immunity measurements for the reference ECU.

The example results show that 100BASE-T1/OABR fulfills the EMC requirements of many[3] car manufacturers. Nevertheless, there are specific applications and harness routes inside a car, in which the shown limit lines are not sufficient. This can be the case if the Ethernet (or CAN or FlexRay) cable needs to pass a sensitive antenna system in

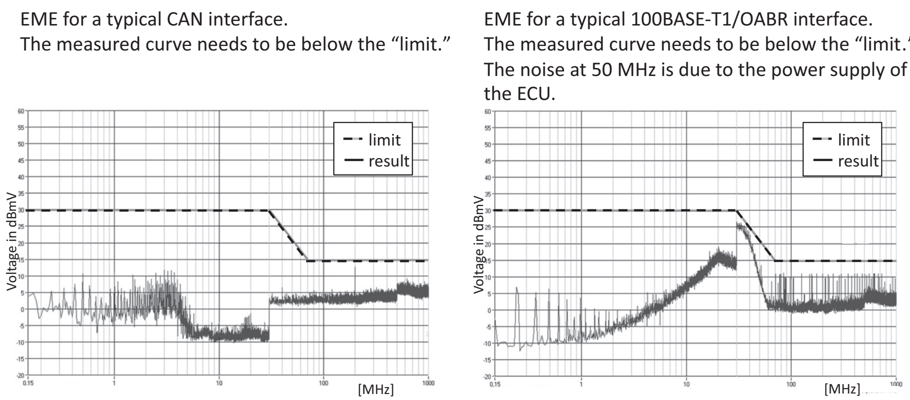

Figure 4.7 Example results for stripline measurements according to ISO 11452–5.

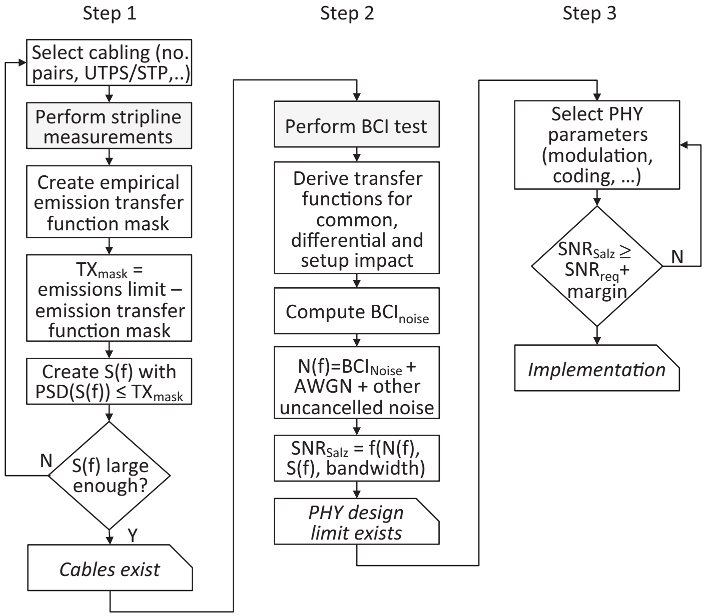

Figure 4.8 PHY specification development process based on EMC measurements and limit lines [17] [11] [16].

close vicinity. Even the use of shielded cables can be critical in these environments and sometimes ferrites are added in addition to the shield in order to meet the requirements. The advantage Ethernet has in this case over other systems like CAN or FlexRay is that Automotive Ethernet is switched with P2P connections only. This means that not a whole bus with all participants needs to receive the extra measures, but just the one link that passes such a sensitive area. For Ethernet-based communication it is not relevant whether one link attached to a switch is shielded or unshielded or optical or carries a different data rate for that matter. The overall costs of the system can be optimized accordingly (see also Chapter 6).

4.1.3.2 EMC Results for 1000BASE-T1

The EMC requirements for technologies used inside cars are significantly more stringent than the EMC requirements for IT- or CE-devices. When 1000BASE-T1 was being developed at IEEE 802.3, with neither cable type nor number of cable pairs known to start with (see also Sections 4.2.4 and 4.3.2.1), the EMC requirements became central to the development process. The project had to start with an EMC-based feasibility study in respect to the cabling type usable. This had not been done before, and therefore also the methodology was yet to be proven and established. This section will give an overview on the process used. Its basic elements are depicted in Figure 4.8.

It is essential for any PHY development to know what channel the PHY is going to be used with. The 1000BASE-T1 project had not yet decided on a channel, but there was a preference: A single pair of Unshielded Twisted Pair (UTP) cables, i.e., Unshielded Twisted Single Pair (UTSP). However, people had been skeptical that the 100 Mbps PHY, 100BASE-T1/OABR, would be usable with UTSP. Now, it was desired to transmit a tenfold data rate over similar cables. The methodology thus consisted of three main steps:

1 Prove that cables and connectors (can) exist that meet the EMC emission criteria while allowing for enough power to be transmitted (to ensure EMC immunity).
2 Use those cables and connectors to derive the (EMC) noise the system has to cope with.
3 Define the PHY parameters such that with the power, noise and resulting bandwidth they leave enough margin for the (always imperfect) implementation (see Section 4.3.2.1 for details).

The main task behind step 1 was to find a suitable transmit power spectrum mask TX_{mask} that in return would allow to set the power of the transmit signal and its PSD. The procedure to obtain this mask (see [11]) is essentially based on the direct correlation between the emissions and the transmit power spectrum TX_{power} as well as the emissions transfer function (see Equation 4.1). This correlation was not known upfront, but had to be proven first [11].

$$\begin{aligned} &\text{Emissions } [\text{dB}\mu\text{V}] = TX_{\text{power}} \; [\text{dB}\mu\text{V}] + \text{Emissions Transfer Function } [\text{dB}] \\ &\Rightarrow TX_{\text{power}} \; [\text{dB}\mu\text{V}] = \text{Emissions } [\text{dB}\mu\text{V}] - \text{Emissions Transfer Function } [\text{dB}] \\ &\Rightarrow TX_{\text{mask}} \; [\text{dB}\mu\text{V}] = \text{Emissions}_{\text{limit}} \; [\text{dB}\mu\text{V}] - \text{Emissions Transfer Function}_{\text{mask}} \; [\text{dB}] \end{aligned} \quad (4.1)$$

So, first stripline measurements were performed with various cables and of those with promising emissions behavior the transfer function was derived. Reference [11] shows example stripline measurements of various such cables. From these results an emission transfer function mask was derived empirically such that the measured emissions stayed below. The difference between the BMW stripline limit of 15 dBμV and the emission transfer function mask was used to obtain the transmission power spectrum mask (TX_{mask}) of the differential signal. Reference [11] gives an example of a the PSD of a signal $S(f)$ with 1 V peak to peak (Vpp) that is well below the TX_{mask} limit. This proved that UTSP cabling was feasible for 1000BASE-T1. Note, that the upper transmit power mask defined in the 1000BASE-T1 specification [12] actually allows for somewhat more power, approximately up to 1.4 Vpp.

The second step addresses the noise the channel is susceptible to, which in automotive needs to include the noise caused by EMC. In differential systems like Automotive Ethernet, the noise is the result of nonideality and imbalances in the channel. Because of these imperfections, the interference has a different impact on one wire than the other in a pair. Therefore an interference can no longer be completely cancelled out and results in common mode as well as differential mode noise. Reference [13] proposes to use the Bulk Current Injection (BCI) immunity test setup in order to quantify and measure the common mode and differential mode transfer functions H_{CM} and H_{DM} of the cables found suitable with the emission tests. Key to computing the common mode and differential mode noise V_{CM} and V_{DM} for a given BCI test profile I_{BCI} (see Equation 4.2), is the correct mitigation of the effects the test setup (test heads, clamp, termination [14] . . .) has on the results (reflected in H_{CIP}).

$$V_{CM}(f)\ [\text{mV}] = I_{BCI}(f)\ [\text{mA}]\ Z_{CM}(f)\ [\Omega]$$
$$\text{with} \quad Z_{CM}(f)\ [\Omega] = \frac{50\Omega}{\sqrt{2}} \left| \frac{H_{CM}(f)}{H_{CIP}(f)} \right|$$
$$V_{DM}(f)\ [\text{mV}] = I_{BCI}(f)\ [\text{mA}]\ Z_{DM}(f)\ [\Omega]$$
$$\text{with} \quad Z_{DM}(f)\ [\Omega] = 50\Omega\ \sqrt{2} \left| \frac{H_{DM}(f)}{H_{CIP}(f)} \right| \tag{4.2}$$

The such obtained EMC noise for the existing (prototype) cables can then be added to the Additive White Gaussian Noise (AWGN) and other uncanceled noise sources to obtain the overall noise $N(f)$. With these inputs it is possible to calculate the Salz SNR as a function of the bandwidth W (see, e.g., [15] or Equation 4.3). The Salz SNR represents the SNR value that is the theoretically achievable with an infinite length equalizer and can be set as a maximum bound. Decision on the actual PHY parameters can be based on this value with the consideration of respective implementation margins [16] and the effects it has on the bandwidth W. For details on the actual PHY parameters selected, see Section 4.3.2.1.

$$\text{SNR}_{\text{Salz}}(W) = 10\log_{10} e^{\frac{1}{W}\int_0^W \log_e\left(1+\frac{S(f)}{N(f)}\right)df} \quad [\text{dB}] \tag{4.3}$$

4.1.4 ElectroStatic Discharge (ESD)

In case of ESD, a short, high-voltage impulse is caused by a spark at an electronic device owing to a large electric difference between the electronic device and the touching entity. Under disadvantageous circumstances the high-voltage discharge can damage the device. Especially field-effect transistors are susceptible to such damage [18]. The cause for the electric potential difference is generally a triboelectric effect or electric induction. A well-known example is walking on a carpet, which can charge a human up to 30,000 V.

In automotive, ESD has many facets and needs to be considered in the whole chain from ESD in the assembly, to ESD caused by service staff, to ESD caused by passengers. To test the robustness of electronic devices to ESD, four ESD-simulation models have been introduced with ISO 10605, which also describes the respective test methods for road vehicles. A good summary on the models can be found in [19]:

- The **Human Body Model (HBM)** models the discharge of an electrostatically charged person when touching an electronic hardware element. The induced current is assumed to pass between different pins of the touched hardware in question.

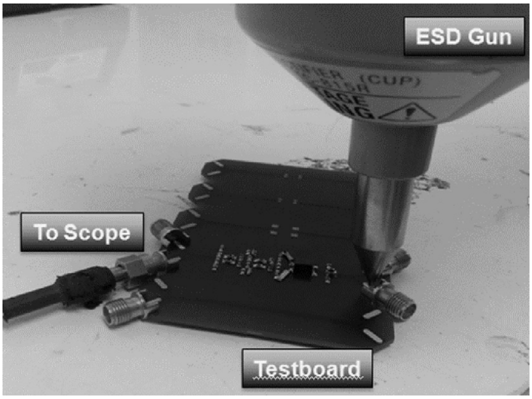

Figure 4.9 Example of a test setup that measures the voltage at the 100BASE-T1/OABR PHY pins in case of ESD at the entrance of the test board. Photograph by Tim Puls, Semtech.

- The **Machine Model (MM)** is similar to the HBM, but instead of a person an electrostatically charged machine discharges in contact with the electronic hardware element. Like in case of the HBM the induced current is assumed to pass between different pins of the touched element.
- The **Charged Device Model (CDM)** is fundamentally different from the HBM and the MM. In case of the CDM the whole electronic device is assumed to have been electronically charged and suddenly discharged against a low resistance electrode. In this case, no current passes through the discharging device.
- In the **Field induced Charge Device Model (FCDM)** also a charged electronic device is assumed to suddenly discharge. The difference is, that the charging happened in an electric field or via electric load shift.

For enabling Automotive Ethernet this means that all transceiver semiconductors need to be tested accordingly; individually and in the respective application. For the semiconductors there is nothing Ethernet specific. Provided they have been designed with standard process technologies, the same tests as for any other automotive semiconductor can be performed, which is part of the AEC-Q100 qualification and recommended design process.

In case of the integration of a part into an ECU, the situation is not quite as straight forward. The key question, how much of the 8 kV discharged at the outside connector contacts of an ECU (see ISO 10605) actually reaches the transceiver chip. This is of particular interest, because transceiver chips are per definition connected to the outside of an ECU and because the continuous miniaturization in the semiconductor industry makes ICs ever more sensitive to ESD. It can be expected that those portions of the Printed Circuit Board (PCB) that lie between the connector and the transceiver chip somewhat reduce the voltage; in case of 100BASE-T1/OABR these are a CMC, a coupling capacitor, and potentially other filter elements. The question is, and this also depends on the particular PCB design, how much does the PCB reduce the discharge voltage? Figure 4.9 shows the example of a respective test setup.

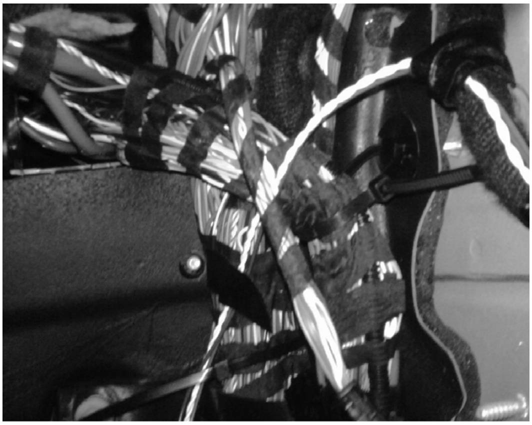

Figure 4.10 An impression of the complexity of a car's harness.

In the case of the BMW test boards, about 500 V of the 8 kV reached the transceiver pins. If the ICs used cannot handle this voltage, additional ESD protection elements need to be added. These elements in return have to be dimensioned such that their parasitic capacitor does not influence the transmission channel (too much). A generic answer or recommendation on how to handle ESD protection for Automotive Ethernet is not possible. In the best case, the PCB design and Ethernet transceiver can handle the residual ESD voltage. In the worst case, the additional ESD protection has to be added.

4.2 The Automotive Communication Channel

The (automotive) communication channel generally constitutes of the cables and connectors used in the wiring harness. Figure 4.10 gives an impression of how complex the wiring harness can be in automotive. Not for nothing is the harness the third heaviest and third expensive part of a car (after engine and chassis; see, e.g., [20]). Ground connections throughout the car are not less complex: Because of new compound materials used, the body is no longer one huge, conducting sheet of metal. Furthermore, the harness passes through separations in order to reach doors, the booth, or the engine compartment.

Harnesses are manufactured in a large variety: For every car model and depending on the options the customer selected, a different harness is being built. In every harness, a large number of cables with different use and functions are in closest proximity to each other, which means that the possibility of crosstalk and asymmetries cannot be neglected. Also, a harness uses components from various suppliers. A harness consists of many different types of cables and connectors: UTP cables are often twisted and connected as part of the harness manufacturing. Shielded, coax, and optical cables are delivered premanufactured with connectors. All these aspects need to be taken into consideration when defining the channel. The following subsections explain the framework for the Automotive Ethernet channel and the parameters used. The complete channel

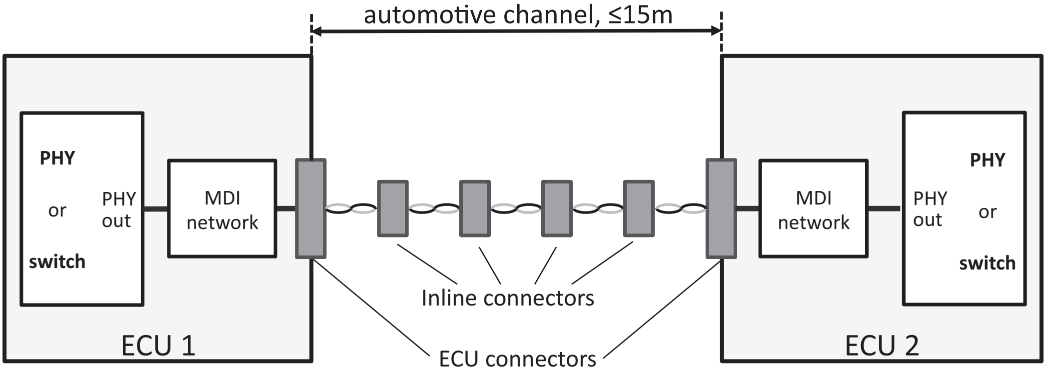

Figure 4.11 Ethernet transmission elements on PHY level identifying the communication channel [23].

description for 100BASE-T1/OABR can be found in [21] and [22]. The 1000BASE-T1 channel is described in [12].

4.2.1 Channel Framework

Figure 4.11 shows the different elements needed for two ECUs to be able to communicate on PHY level. Before discussing the channel parameters, it is essential to define the elements the channel comprises of and to identify the elements that are part of the PHY transmission but not part of the channel. Figure 4.11 shows that the parts of the ECU – the PHY transceiver IC, the Media-Dependent Interface (MDI) network, and the connector parts attached to the ECU – are not part of the channel. Instead, the channel consist the cable, up to four inline connectors and the end connector parts attached to the cables. The standard maximum channel length for automotive is 15 m.[4]

As an example, Figure 4.12 shows the elements of the MDI network for 100BASE-T1/OABR and the location of the MDI. The MDI is where the media changes from PCB to wire. The MDI network is in principle independent from the function of the channel. Channel and MDI network are designed to meet certain requirements and limit lines and require that the respective other part meets theirs as well. If the goal is to further

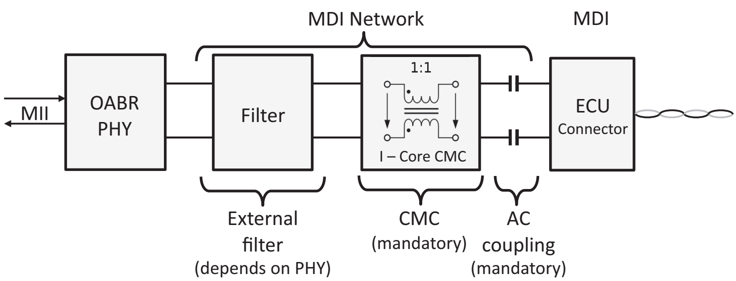

Figure 4.12 Elements of the MDI network for 100BASE-T1/OABR Ethernet [24].

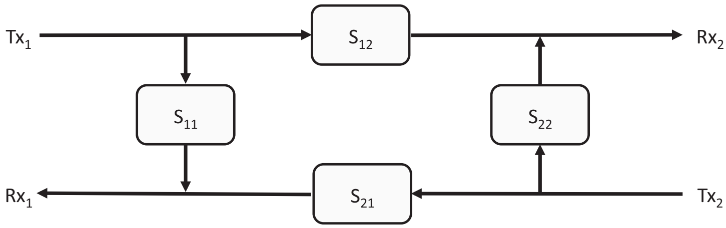

Figure 4.13 Relationship between S-parameters and receiver ports in a single-pair Ethernet system.

improve, e.g., the MDI performance, this simply improves the overall performance without having any (negative or positive) impact on the channel performance. This leaves room for manufacturers to optimize their parts.

The MDI network consists of three major parts:

- **AC coupling (DC blocking):** Some coupling mechanism is mandatory in order to suppress DC from the transmission. Section 4.5.2 explains in more detail why it is possible to use capacitors instead of transformers for 100BASE-T1/OABR and 1000BASE-T1. Capacitors are technically sufficient and more cost efficient than transformers.
- **CMC:** The CMC is one of the most important components in the system. Its function, the common mode suppression, is vital for the ElectroMagnetic Immunity (EMI). Section 4.5.2 details, why an I-core CMC can be used with 100BASE-T1. This is desirable, because I-core CMCs allow for fully automated production. As the bandwidth for 1000BASE-T1 is about 10-fold the bandwidth of 100BASE-T1, the 1000BASE-T1 CMC needs to perform wide band suppression at 10 dB better performance. For details of the 1000BASE-T1 CMC specification see [25].
- **Filter:** The additional filter in the MDI network performs spectral shaping in order to improve the EMC performance. The use of the filter depends on whether the actually used transceiver chip makes additional filtering necessary or not. This part would also comprise an ESD suppression circuit, if the specific PHY semiconductor requires its use (see also Section 4.1.4).

4.2.2 Channel Parameters

Important values to characterize the transmission channel are the Scattering parameters (S-parameters) [26]. In principle, S-parameters can be used to describe the electrical behavior of any linear electrical network that is steadily stimulated by electrical signals. As they are particularly suitable for microwave engineering, S-parameters are practical values to determine the channel for Ethernet systems. Figure 4.13 shows the relationship between the S-parameters and a communication system that, like 100BASE-T1 and 1000BASE-T1, uses a single wire pair for transmission only. S-parameter models can be extended to describe (Ethernet) systems that require more cable pairs. As neither 100BASE-T1 nor 1000BASE-T1 need more pairs, it is not discussed here. The

S-parameters for the one pair system are S_{11}, S_{12}, S_{21}, and S_{22}. Their actual characteristics for a given channel can be measured with a network analyzer, whose output results can then be matched with the allowed limit lines.

For robustness reasons, even communication systems with data rates significantly below 100 Mbps generally use differential signaling (see also Section 2.2), because the communication channel does not only carry the differential signal, but also interference with common mode characteristics. The whole idea of the differential signal is that the common mode interference is suppressed at the receiver, when both parts of the differential signal are recombined to one signal, i.e., converted back from a differential to a common receive signal. This works comprehensively when the interference is symmetric and it is thus a goal for Automotive Ethernet to have the channel set up as symmetric as possible. Unfortunately, every however carefully designed network can have some asymmetries, be it slightly different lengths of the two wires within the UTSP cable or an unavoidable neighboring power supply in a multipin connector. The EMC performance of a system is significantly influenced by just these asymmetries. In order to be able to define limits for this as well, the S-parameters thus distinguish between values for the differential transmission performance "dd" and values for the common mode to differential mode "cd" and differential mode to common mode "dc" conversion performance.

The two performance values that are directly identified by the differential performance are the Insertion Loss (IL) and the Return Loss (RL). The IL, i.e., the attenuation the signal experiences when traveling from transmitter to receiver, can be defined as $IL = f(S_{dd,12}, f) \approx f(S_{dd,21}, f)$. The RL, i.e., echo strength a received signal is impaired by from reflections of its own transmission, can be defined as $RL_1 = f(S_{dd,11},f)$ and $RL_2 = f(S_{dd,22},f)$.

Furthermore, parameters are needed that describe the symmetry (also called balance) between two wires of one cable pair that is so important for the EMC. Originally, [23] used the Transverse Conversion Loss (TCL) and the Equal Level Transverse Conversion Transfer Loss (ELTCTL) for this purpose. The TCL measures the echo of the common mode signal as a function of $S_{cd,11}/S_{cd,22}$. The ELTCTL measures the common mode to differential mode conversion in relation to the use signal as a function of $S_{cd,12} - S_{dd,12}$ and $S_{cd,21} - S_{dd,21}$. Newer publications, e.g., [22], [12], and [21], use the Longitudinal Conversion Loss (LCL, $S_{dc,11}/S_{dc,22}$) and the Longitudinal Conversion Transmission Loss (LCTL, $S_{cd,12}/S_{cd,21}$) instead. LCL and TCL as well as LCTL and TCTL provide same technical information [27]. TCTL replaced ELTCTL because the "Equal Level" turned out to be hard to maintain in real systems that always experience some kind of attenuation. Especially in case of channels with high attenuation, this can lead to results that do not represent the actual situation [28].

Further important channel parameters are the impedance and parameters related to crosstalk, i.e., the interference caused by surrounding cable pairs (see also Section 4.1.1). With both 100BASE-T1 and 1000BASE-T1 being single pair, all crosstalk comes from sources unknown and thus "alien." The limit lines for the Alien Near-End Cross Talk (ANEXT) are described by the Power Sum Alien Near-End Cross Talk

(PSANEXT). To describe the AFEXT impact on the far end the Power Sum Alien Attenuation to Crosstalk Ratio Far-end (PSAACRF) is used.

Having limit lines for the channel available is not only crucial for being able to develop the PHY transceiver solutions. From the channel limit lines, it is possible to deduct requirements that allow manufacturing a harness and its components in a way that meets the automotive quality requirements and to identify optimization potential. This is done by reverse engineering, i.e., measuring the S-parameters of certain samples against the limit lines.

4.2.3 The 100BASE-T1/OABR Channel

When developing a communication technology, the technical properties of the channel the technology is meant to run on, are generally the first parameters that need to be known. With those parameters available, it is possible to design the optimum system for this channel.

BroadR-Reach, the technology 100BASE-T1 was specified from, was originally not designed for automotive use, but was found suitable nevertheless. Contrary to the normal development sequence, 100BASE-T1/OABR was thus developed and adopted in the automotive industry without exact knowledge of the channel parameters. The requirements were: To be able to use UTP cabling as well as standard automotive connectors and that the system would show a stable performance fulfilling the strict automotive EMC requirements. The solution did show the requested behavior. However, it is important to know margins and tolerances in order to optimize ECU design. The channel parameters are additionally needed for new semiconductor vendors in their product design process. In consequence, the 100BASE-T1/OABR channel limit lines described in [23] and [22] were defined in hindsight.

Figure 4.14 shows example measurement results of channel parameters for a 100BASE-T1/OABR channel as defined in [23]. Additionally the impedance required is 100 Ohm $\pm$10%.

Various aspects of the actual wiring harness scenario can impact its performance: The ambient temperature, the conductor length, material, and size, the insulation dielectric, color and thickness, the number of inline connectors, the wire twist rate, the consistency of lay length, the untwist length at connectors, loops in the cable, whether the cable is jacketed or not [29], as well as pinning in case of multipin connectors. Thus design criteria need to be derived for use in cars in respect to these elements.

Figure 4.15 visualizes an example of one design criteria derived from evaluating actual cable measurements against the limit lines: the maximum untwist area 100BASE-T1/OABR cables allow for when being connected (30 mm). This untwist length consists of two parts that are inside the connector, in which twisting is generally not possible, and the area just before the connector. Automotive quality requires that every connection in a 100BASE-T1/OABR link in the harness has an untwist area shorter than defined in the requirement depicted in Figure 4.15 (19 mm). In order not to need to measure the areas with every connector attached, the right, automated production processes need to be in

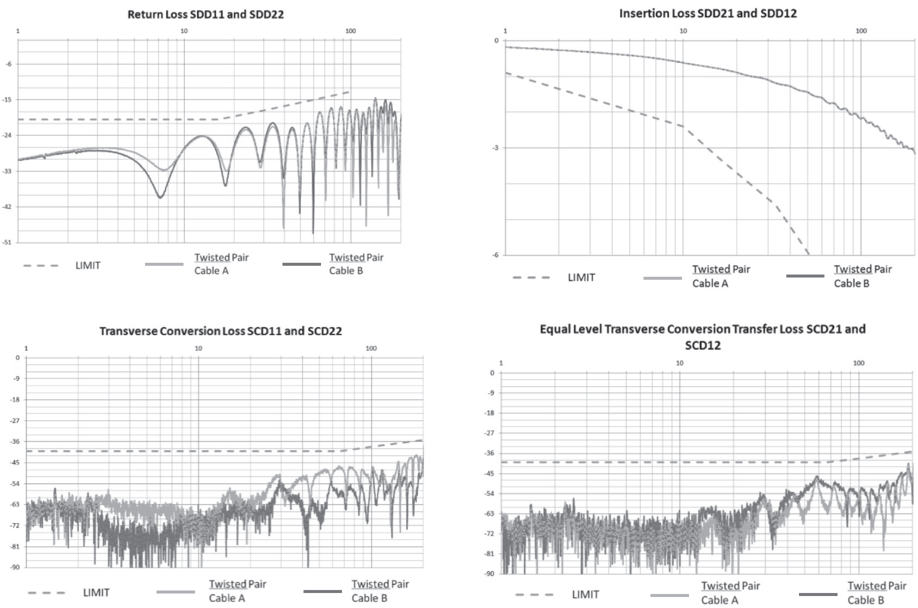

Figure 4.14 Example S-parameter measurements for a 100BASE-T1/OABR channel with two cables. For each measurement, one cable was measured in both directions with a four-port network analyzer. Both cables measured are automotive qualified.

place. In case of 100BASE-T1/OABR, machines can assure the twisting close up to the connector. Such automated process is "safe" and thus sufficient. Other requirements that can be deducted from the channel definition refer, e.g., to the dielectric material used for the cable insulation. Note that BMW uses standard Micro Quadlock System (MQS) connectors for the 100BASE-T1/OABR links.

4.2.4 The 1000BASE-T1/RTPGE Channel

When the IEEE802.3bp Reduced Twisted Pair Gigabit Ethernet (RTPGE) Task Force (TF) took up its work in January 2013, it first had to define the channel. This was unusual. Most other IEEE 802.3 PHY projects select existing cables (or existing cable

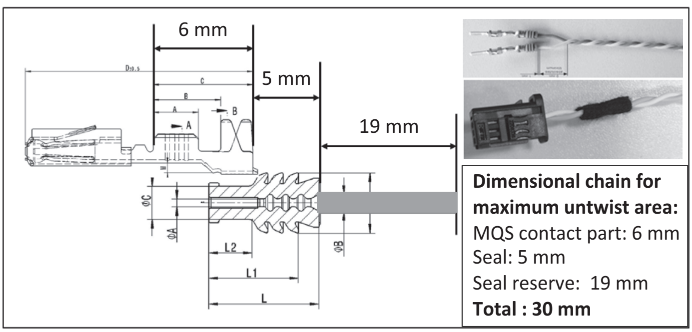

Figure 4.15 Maximum untwist area for OABR in case of connector attachment [30].

definition projects[5]) as target channels directly with the objectives. In automotive neither channel nor cables had been defined in a respective manner; also not for 100BASE-T1/OABR, for which the OPEN Alliance was just then completing the channel definition (see also Section 4.2.3) The TF thus first had to develop and agree on the respective limit lines for RTPGE. This posed two challenges:

1 The automotive environment differs from the IT environment especially in the EMC performance. Developing the correct limit lines requires detailed understanding in the TF of the requirements and test setups relevant for automotive.
2 There is a trade-off between the capabilities the PHY needs to provide and the quality of the cables and connectors in between. For example, the smaller the attenuation of a signal during the transmission, the smaller the requirements on the PHY in respect to equalization and echo cancellation, but the better and likely more expensive the cabling needs to be. The larger the allowed attenuation during a transmission, the less effort is required in the cabling, but the larger the effort and thus expense of the PHY in order to extract the correct information from the received signal, because larger attenuation also means larger susceptibility to interference. If a predefined cable is selected, this trade-off will likely not cause discussions. Without a predefined cable, discussions on which side can accept what "burden" are more likely. The solution might require several iterations, each requiring proof on the limits that can really be reached.

The first most controversial topic concerning the channel had been agreed on during the preceding Study Group (SG) phase: The link length. For an Ethernet link in passenger cars 3.5 m length is a good average value (see, e.g., [31]). A 10 m link can easily connect a sensor in the right corner of the front bumper with an Electronic Control Unit (ECU) that sits in the left part of the booth, even in a long passenger car. When including vans and light trucks and connecting cameras at exposed ends, 15 m is still sufficient, while long haul busses and large trucks can need up to 30 m [32]. For the industrial automation industry, 100 m is a standard link length requirement [33]. So, while it is attractive to address many applications, the volume, however, is in the shorter automotive links. A cost-efficient, successful technology for a volume market will more likely be adapted for a more challenging market, than a more expensive technology to a volume market. As a result, RTPGE was to be designed for a 15 m link segment with up to four inline connectors. Once this was achieved the same PHY was to be used to investigate how much longer the link segment can be – the objectives proposed an optional 40 m link segment – with better cabling [34].

The second most controversial topic was the number of twisted pairs to use for the cables. RTPGE meant reducing the number of pairs to fewer than four – the number of pairs used for 1000BASE-T (see also Section 4.3.1.1). For cost efficiency reasons, Unshielded Twisted Single Pair (UTSP) was the preferred target cabling of the car manufacturers. If the environment becomes more challenging, this, in principle, allows car manufacturers to move first to the next more costly option of coax cables and then to shielded cables. Coax was not officially addressed in the IEEE project. However, the use

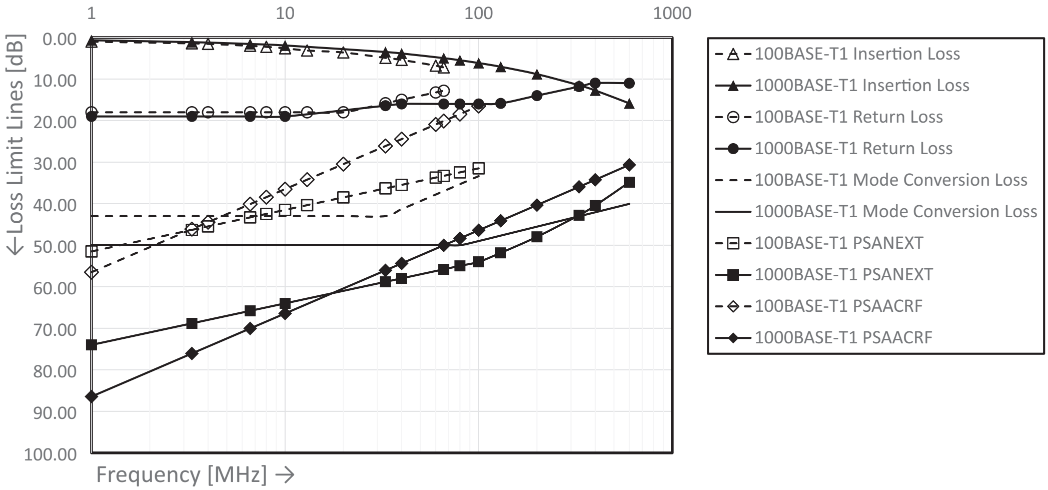

Figure 4.16 Comparison of channel limit lines for 100BASE-T1 and 1000BASE-T1.

of one pair of twisted cables in principle allows for this step, while the selection of two pairs would have prohibited it. The decision in the TF was thus to work with a single pair of cables. Only if a solution for one pair was impossible to find, were two pairs to be considered as an alternative [35]. In the end, 1000BASE-T1 was designed for UTSP and the naming 1000BASE-T1 for the IEEE 802.3bp/RTPGE technology somewhat reflects the outcome.

The resulting channel limit lines are shown in Figure 4.16. Next to the impedance, IL, RL, PSANEXT, PSAACRF and the common mode to differential mode conversion loss (see [36] and [37] for details), the TF investigated wire gauge and temperature impact (see, e.g., [38], [36]) as well. Finally, cable and connector companies provided measurements as proof that these requirements can be met.

Figure 4.16 also compares the limit lines for 1000BASE-T1 with the limit lines for 100BASE-T1. The most obvious observation is that the 1000BASE-T1 channel has to deal with a tenfold bandwidth. However, in respect to IL and RL this results in more stringent performance requirements at higher frequencies “only,” while the requirements in the lower frequencies are very similar. This is different for the mode conversion and crosstalk values. The tenfold bandwidth means that the signal is significantly more susceptible to interference. In order to maintain the same Signal-to-Noise-Ratio (SNR) without increasing the signal, the noise has to be smaller and the channel thus needs to suppress the interference better. For the mode conversion this means about 7 dB, for the ANEXT (PSANEXT) about 22.5 dB and for the AFEXT (PSACRF) about 30 dB better performance, also at the lower frequencies. In order to achieve these values, while still deploying UTSP cables, it has been proposed to use jacketed cables . The jacket ensures a certain physical distance to the next neighboring cables, while at the same time giving better stability to the twist and with that improving the balance, even in areas where the cable is bent (regularly). This additionally means that also the used connectors have to

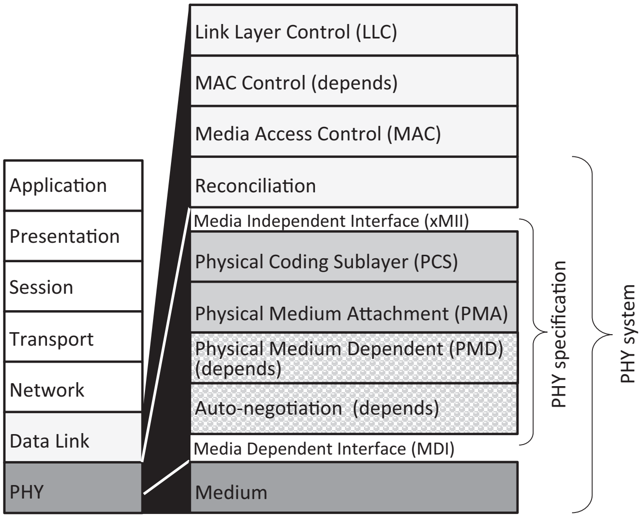

Figure 4.17 Overview on PHY sublayers in respect to OSI levels as referenced for various IEEE802.3 specifications.

provide a better performance, with shorter untwist areas and better physical separation to the neighboring pins than the (n)MQS connector provides. For details on the harness components and their requirements see [39].

4.3 The Physical Layer (PHY) Technologies

The IEEE 802.3 Ethernet physical layer specifications describe the methods and protocols needed in order to allow for a manufacturer-independent interoperable communication on PHY level. As visualized in Figure 4.17, the principles described are logically located between the respective Media-Independent Interface (xMII) that connects the PHY with the MAC layer and the Media-Dependent Interface (MDI) that is the interface to the transmission medium, i.e., the channel described in Section 4.2.

The two main sublayers of the PHY are the Physical Coding Sublayer (PCS) and the Physical Medium Attachment (PMA), which are normally implemented in one ASIC. The PCS receives digital data from the xMII and encodes the data into symbols for the consecutive processing by the PMA. It also decodes the received signal into a bit stream ready to be passed to higher layers via the xMII. The PMA has the task to physically prepare a signal for transmission and to prepare the receive signal such that the coded information can be extracted from it by the PCS.

Depending on the technology, more sublayers are added to the PHY. Figure 4.17 shows a layer for autonegotiation, a method to establish and select the best communication capabilities in respect to data rate, duplex mode, and flow control. This is crucial, when the same channel can have different speed grades attached, as happens frequently in the uncoordinated plug & play environments of the Consumer Electronics (CE) and

Information Technology (IT) industries. In the predefined in-vehicle networks, the situation is slightly different and scenarios where plug & play is needed are the exception and not the rue. Because they can occur nevertheless, autonegotiation has been included as an optional feature in 1000BASE-T1 specification (see Section 4.3.2.1) and is therefore part of Figure 4.17. Furthermore, some depictions explaining the Ethernet PHY elements additionally include a Physical Medium-Dependent (PMD) sublayer. The PMD is relevant when for the same transmission standard different media can be used that require (each) additional and different handling of the signal, before it is put on the channel, e.g., if an optical transmission is specified, the PMD defines the translation between the electrical and the optical analog signals. It also sits between PMA and MDI and is part of the 1000BASE-RH specification described in Section 4.3.2.2.

4.3.1 100 Mbps Ethernet

It all started with the IEEE 802.3 1000BASE-T standard. During its development the Broadcom engineers learned to handle the communication challenges that needed to be mastered for such a high data rate transmission. So when Ethernet in the First Mile (EFM) was being developed at IEEE, Broadcom reused some of the basic principles of 1000BASE-T for a suitable solution: Instead of four pairs of wiring one pair was used and the channel coding was made more robust, so that it was possible to transmit 100 Mbps data over a worse, i.e., longer channel. IEEE standardized a different solution for EFM, while Broadcom proposed their technology for EFM in China [40]. When BMW was looking for an Ethernet solution suitable for automotive in 2007 another interesting use case was found for the Broadcom technology. This technology was named BroadR-Reach at the time, then published by the OPEN Alliance and called OPEN Alliance BroadR-Reach (OABR) in 2011, before finally being ratified as IEEE 802.3bw standard in 2015, naming the technology 100BASE-T1. To create an understanding for 100BASE-T1, this section first addresses in Section 4.3.1.1 some fundamentals of 1000BASE-T, before the 100BASE-T1 technology is explained in Section 4.3.1.2.

However, there is more to 100 Mbps Ethernet in automotive. After the proof had been provided that it is possible to transmit Ethernet packets at 100 Mbps over Unshielded Twisted Pair (UTP) cabling in the automotive environment with BroadR-Reach, other companies started to provide different, i.e., incompatible solutions to the same end. For example, another semiconductor vendor presented a solution to BMW as early as 2009 that also used a single pair UTP (Unshielded Twisted Single Pair, UTSP) cable. Even if the vendor decided not to pursue its technology, it was an important milestone for BMW. In 2009, the market was not ready for Automotive Ethernet. Among most of the decision makers in the automotive industry skepticism prevailed on the technical feasibility and on the need of such a technology. Therefore, there was also no obvious market prospect for the decision makers in the semiconductor industry.

The authors therefore doubted at the time and still doubt that starting a public standardization of BroadR-Reach – which requires faith in the technical feasibility and a market prospect – in 2009 or earlier, would have been successful. Additionally, a solution was needed fast. Else the targeted SOP in 2013, and with it potentially the complete market development of Automotive Ethernet, could have been missed. The other solutions that were being proposed were proof that multiple semiconductor vendors would be able to handle automotive UTSP Ethernet. This, in the end, motivated BMW to go ahead with enabling and qualifying BroadR-Reach as well as establishing a multisourcing strategy for the technology (see also Section 3.3 and following).

Section 4.3.1.3 addresses one alternative developed, which uses UTP cabling with 100BASE-TX for automotive use. It can provide some advantages, e.g., in the context of Diagnosis-over-IP (DoIP), which has been standardized to use 100BASE-TX in ISO 13400. Last but not least, Section 4.3.1.4 shows the flexibility that can be achieved with an Ethernet network, because of its strict layering approach. Section 4.3.1.4 shows how the MII interface allows for quite different approaches to transmit 100 Mbps Ethernet frames.

4.3.1.1 The Reference for 100BASE-T1/OABR: IEEE 802.3 1000BASE-T

One of the goals during the development of 1000BASE-T was to meet the same Federal Communications Commission (FCC) class A ElectroMagnetic Compatibility (EMC) requirements as has been done for 100BASE-TX [41]. As a result, 1000BASE-T uses more or less the same spectrum as 100BASE-TX (see also Figure 4.23 and Table 4.10). With the technologies available at the time, this was not possible to achieve with one or two pairs of wires only. Instead a cable with four pairs, CAT 5e, was selected. In order to keep the bandwidth required per cable pair low, it was necessary to implement a "true" full-duplex mode, i.e., to transmit and receive on the same wire. This in return led to the use of echo cancellation and hybrids. 100BASE-TX, in contrast, uses one of its two cable pairs for transmission and one for reception.[6] For 1000BASE-T autonegotiation was made mandatory. However, there are two reasons why autonegotiation is not discussed with 100BASE-T1. A) 100 Mbps Ethernet is the first speed grade used in automotive. Data rate and duplex mode are unambiguous for 100BASE-T1. B) With the preset network inside a car, it is rare that autonegotiation can provide a benefit. This is quite different from the plug & play in the IT and CE industries.

In the following 1000BASE-T elements important for the understanding of 100BASE-T1 are being described:

- **Hybrid:** A hybrid is used in case data is transmitted and received simultaneously over the same wire pair ("true" full-duplex operation). Its function is to cancel the transmitted signal from the signal at its pins at $t = 0$ in order to obtain the received signal such that the dynamic range of the receiver can be relaxed. Note that in the original CSMA/CD Ethernet, the transceiver only either transmitted or received data, as the media was shared among all participants. A hybrid would thus have been useless. With 1000BASE-T transmitting and receiving on the same wire pair, the intention is to use 1000BASE-T in a switched Ethernet network with P2P links only, even if the original standard still considers the half-duplex mode [42].

  Figure 4.18 vizualizes a hybrid circuitry and its use in 1000BASE-T and 100BASE-T1. The differential transmit signal A is led via two paths to ground; one which comprises of the two resistors R1 and R3 and on the other which comprises of the two

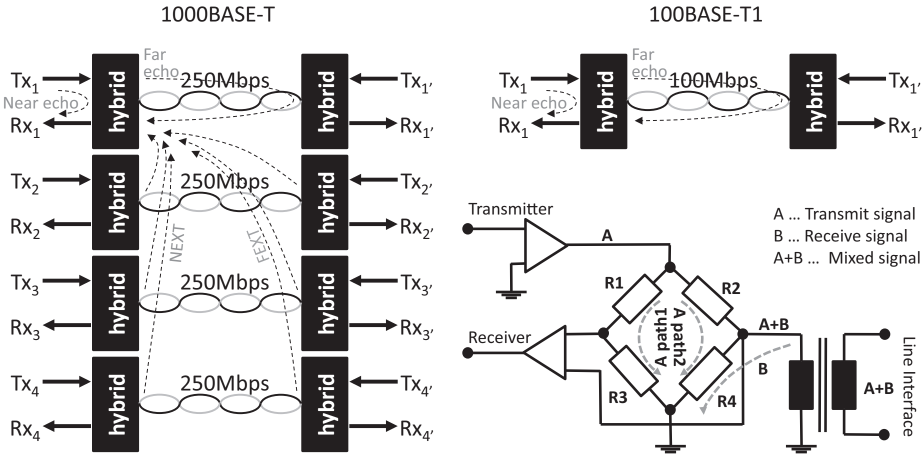

Figure 4.18 IEEE 802.3 1000BASE-T and 100BASE-T1 hybrid use and inner system interference shown for the receive signal Rx_1 as an example [42] [41] [43].

resistors R2 and R4. The output of the receiver sees only the transmit signal A that is passed via resistor R2. On the receive side, the differential receive signal B is connected to the middle of this bridge. This means that the receive signal is terminated via R4, but also that the receive signal, at least theoretically, does not see the transmit signal. The receiver path is completely decoupled from the transmit path. In a real system, however, the resistors are never perfectly matched and a perfect balance cannot attained for the hybrid circuit. Therefore, nonideal hybrid cancellation will cause a small leakage signal to occur at the receiver input. The line interface of the system sees the mixed signal A+B.

- **PCS:** The 1000BASE-T1 PCS processes the data received from the Gigabit MII (GMII) such that it passes "optimal" Four-Dimensional Five Level Pulse Amplitude Modulation (4D-PAM5) coded symbols on to the PMA. At the same time it receives coded symbols from which it extracts the information to send via the GMII to higher layers. The different elements of the 1000BASE-T PCS are shown in Figure 4.19.

  A main feature of the 1000BASE-T PCS is the use of a pseudo-random bit "side stream." The goal of this side stream is that the transmitted symbols are as different

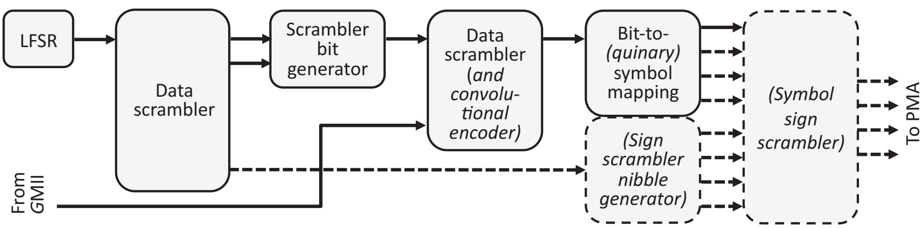

Figure 4.19 1000BASE-T PCS elements and their relation (dashed lines and italics indicate the differences to 100BASE-T1) [44] [45].

from each other as possible and that the signal energy is evenly spread over the available frequency band. This improves the EMC behavior). EME on the wire are reduced and the transmission is less susceptible to EMI. As different pseudo-random bit streams are used for transmit and receive side – one will be master, the other slave – this also ensures that the receive and transmit signals are different enough to not continuously cancel each other out. Furthermore, it prevents that the same symbol is transmitted too many times in a row, which might otherwise create a common signal on the channel.

To generate the side stream, first of all a Linear Feedback Shift Register (LFSR) is used with the master and slave polynomials shown in Equation 4.4.

$$g_M(x) = 1 \oplus x^{13} \oplus x^{33}$$
$$g_S(x) = 1 \oplus x^{20} \oplus x^{33} \qquad (4.4)$$

A following data scrambler generates three bit streams, of which two are processed further with the scrambler bit generator into one bit stream that includes an odd and even timing distinction. This resulting bit stream is used for control, idle, and training mode as well for direct scrambling with the transmission data TXD_n coming from the GMII. Each TXD_n bit is combined with one bit from the random bit stream via a bitwise XOR function between the transmission data and the random bit [46]. With help of a convolutional encoder for 8 bit code word a ninth bit is generated. In the following bit to quinary symbol mapping the 9 bits are divided into subsets, which each determine in different ways the (group of) 4D-PAM5 symbols selected for the four wire pairs (see, e.g., [47]). To ensure enough randomization each set of 4D-PAM5 symbols is scrambled with data derived from the first data scrambler following the LFSR.

The symbol mapping used is quite sophisticated. The 4D-PAM5 uses five voltage levels [1 V, −0.5 V, 0 V, 0.5 V, 1 V] on four wire pairs. This theoretically allows for $5^4 = 625$ different symbols, while the 8-bit code words require $2^8 = 256$ different symbols only. 1000BASE-T1 exploits this for transmitting additional control information but also for making the transmission more robust by selecting the symbols only from a specifically reduced set of symbols [44]. The transmission signal experiences attenuation, InterSymbol Interference (ISI), ANEXT, AFEXT, and other interference while being on the channel. With a voltage difference of only 0.5 V between symbols, the signal is more susceptible to noise and thus decoded incorrectly than if the difference was larger. In order to increase the voltage difference between subsequent symbols (see Figure 4.20 for a depiction of the basic principle) while at the same time profiting from five voltage levels, the symbol mapping distinguishes between four sets of symbols, two each derived of the voltage levels [−1 V, 0 V, 1 V] and [−0.5 V, 0.5 V]. From these sets the 4D-PAM5 code words are selected. The methodology applied – of convolutional coding with an extra bit per 8-bit code word and signal mapping by set partitioning – is called Trellis Coded Modulation (TCM) [45]. At the receiver, the PCS blocks are applied in reverse order.

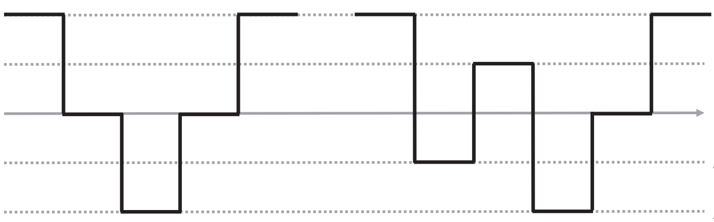

Figure 4.20 Comparison of 100BASE-T1, PAM3 and 1000BASE-T, PAM5 voltage successions.

100BASE-T1 reuses a lot of the methodologies defined for 1000BASE-T (for details see Section 4.3.1.2). It uses the same polynomials and reuses parts of the first data scrambler, of the scrambler bit generator and the second scrambler that exors the random data with the transmission data. As 100BASE-T1 is single pair and only using PAM3 (see Figure 4.20), it has no use for the 1000BASE-T elements implemented to cater for the four channels and the "maximum distanced" 4D-PAM5. These are, e.g., third data random data stream, the sign scrambler nibble generator, the symbol sign scrambler, and the TCM. One other important difference not immediately visible is that for 100BASE-T1 the bit stream used for control, idle, and training mode is not part of the data processing in the side stream, like is the case for 1000BASE-T, but inserted after the symbol mapping (see also Figure 4.25).

- **PMA:** Figure 4.21 shows an example PMA structure, which has the task to prepare 1000BASE-T data for transmission and received data for decoding by the PCS. This entails tasks like clock recovery and reset. As there are four wire pairs the PMA has to be provided four times for the transmit and four times for the receive path.

The incoming transmit data passes first a partial response pulse shape filter with the transmission function 0.75 + 0.25z-1. This filter adds to the multiple of 0.75 of the actual symbol a multiple of 0.25 of the previous symbol in order to reduce the power spectrum of the transmitter and meet the EMC requirements by reducing the EME. The symbols are then converted into analog signals before being overlaid onto the existing signals on the channel by the hybrid.

The received, analog signals are provided by the hybrid. A High Pass Filter (HPF) can follow in order to reduce the required dynamic range of the analog to digital (A/D) conversion, but especially to cut the ISI and with that reduce the number of taps needed in the equalizer later. Feedback from previous received data is used to optimize the power of the receive signal with help of Adaptive Gain Control (AGC) and amplifier, before the signal is actually converted. The now digital signal is then processed in the demodulator, which in the example given contains a Feed Forward Equalizer (FFE), a deskewer, and a (Trellis) decoder. The deskewer aligns the delay differences between the four different pairs and is therefore not needed for 100BASE-T1/OABR. As the cables may have slightly different physical lengths, they can have

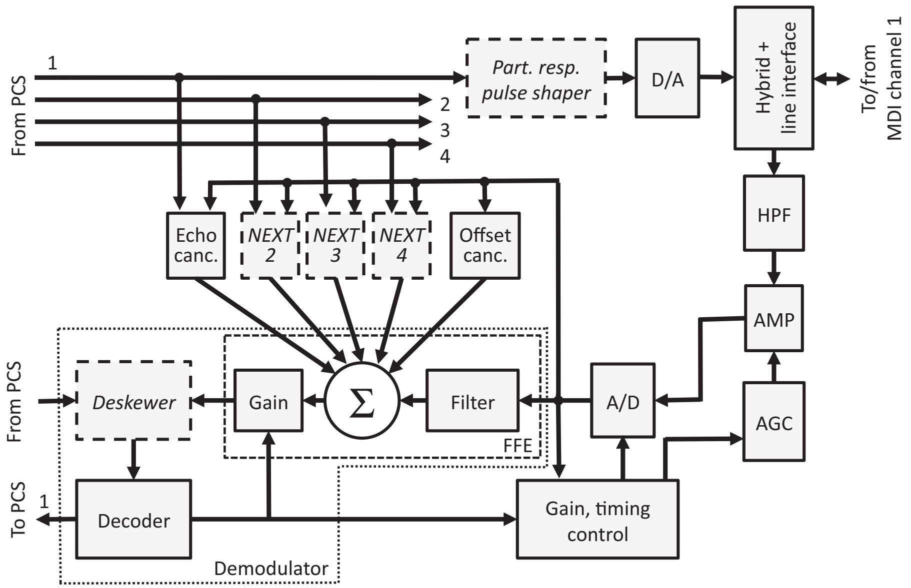

Figure 4.21 Example of a PMA structure for a 1000BASE-T PHY (see, e.g., [44] [41] [48]). Dashed lines and italics mark the blocks not needed for 100BASE-T1.

different propagation times with respect to each other. It requires feedback from the PCS to achieve the right lock in the deskewer.

Inside the FFE, the signal first passes a pulse shaper with the transmission function $g + z^{-1}$. The value of g depends on the cable length, which is deducted from the receive signal strength. Then the inverse partial response is performed in order to reverse the artificial signal spread performed in the transmitter and to ease equalization. The used transfer function $1 + Kz^{-1}$ with $K\,[0,1] \in \mathbb{R}$ is constantly dynamically adapted in order to mitigate disturbances caused by the pulse shaping of the transmitter. It is entirely up to the implementer of the technology to decide on these blocks of the receiver. For example, the example does not contain a Decision Feedback Equalizer (DFE), which might well be seen as a suitable addition.

In the following, the signal is freed from voltage portions resulting from echoes caused on the channel by the own transmit signal and from the nonalien Near-End CrossTalk (NEXT), caused by the transmit signals on the respective other three wire pairs of the 1000BASE-T system. The interference caused by Far-End Cross Talk (FEXT) cannot be known exactly, but could be estimated and then also subtracted. However, the level of FEXT is so low for 1000BASE-T that this is typically not done. Last but not least there is an offset subtraction, which removes the remaining DC caused by an imperfect front-end. The resulting signal is then again amplified and equalized. The following decoder decides which PAM5 symbol must have been received based on the voltage level at its entrance. 1000BASE-T1 requires a Trellis decoder for this, because of the convolutional encoder in the PCS transmit path. Differences between actual and selected voltage level are used to optimize the power of the received signal, as discussed.

Also the A/D receives feedback. The A/D converter sampling time is adaptively controlled by a digital Phase-Locked Loop (PLL), which recovers and tracks the frequency and phase offset for the slave PHY. The master PHY does a similar timing recovery function for the phase offset when the slave transmit clock is frequency locked to the master reference clock. The output signal is passed onto the PCS for final decoding.

The 100BASE-T1 PMA is in principle very similar, albeit it can be somewhat simplified. As is shown in Figure 4.21 by the dashed lines, the 100BASE-T1 transmit data do not pass through a partial response pulse shaper. The receive signal of 100BASE-T1 does not need any of the elements that handle the four pairs of 1000BASE-T1 like the NEXT removal or deskewer. For details on the PMA of 100BASE-T1, see Section 4.3.1.2.

- **Master–slave principle**: Establishing a link between two Ethernet PHYs or link partners requires several steps, which take into account whether a PHY functions as slave or as master. Master and slave PHY definition is being used for two reasons: First, in case of true full-duplex operation, i.e., when data is transmitted and received simultaneously on the same wire pair, master and slave have to be assigned different, predefined scrambling polynomials (see PCS description above). This ensures uncorrelated data and idle streams on the wire. Second, the master and slave distinction is used for the loop-timing concept, which synchronizes the clocks. The master PHY originates a reference transmit clock, which is recovered by the slave PHY receiver for determining its own frequency and phase offsets in respect to the master reference. Then, the slave PHY uses this receive clock to generate its own transmit clock in order to transmit back to the master PHY. At this stage, the master PHY only needs to recover the phase offset in its receiver from the slave signal, as the slave is already using the same frequency as the master. This action completes the loop for timing synchronization between master and slave PHYs. The wide deployment of 1000BASE-T [49] can be seen as an empirical proof for the robustness of this mechanism.

After the power-up, master and slave PHYs go through a handshake process for start-up, sometimes also called "PHY link-acquisition process." This process uses three different signals:

- SEND_Z describes the transmission of zero-code (inactivity or "transmit silent").
- SEND_I describes the transmission of PAM3 idle symbols, which can have the voltage levels [-1 V, 0 V, $+1$ V].
- SEND_N describes the transmission of PAM5 data or idle symbols which can have the voltage levels [-1 V, -0.5 V, 0 V, 0.5 V, $+1$ V].

After the link is enabled the start-up process begins with only the master sending SEND_I idle symbols and the slave staying quiet, i.e., sending SEND_Z. During this time the master trains its echo canceller and the slave synchronizes onto the master clock, adjusts its timing recovery and its equalizer, and locks its scramblers. In the second step, the slave sends idle symbols and trains its echo canceller while the master

Table 4.3 Training phase during start-up in order to converge to minimum errors [50]

Master			
Transmit "idle" (SEND_I)	Transmit "idle" (SEND_I)	Transmit "idle" (SEND_I)	Transmit data (SEND_N)
Adapt echo canceller	Adapt AGC Phase recovery Adapt FFE Lock scrambler	Refine adaptation	
Start-up sequence →			
Slave			
Transmit "silent" (SEND_Z)	Transmit "idle" (SEND_I)	Transmit "idle" (SEND_I)	Transmit data (SEND_N)
Adapt AGC Clock recovery Adapt FFE Lock scrambler	Adapt echo canceller	Refine adaptation	

uses the information to adjust its timing, equalizer, and scrambler. In the third step, the transmitted idle frames of both master and slave are used to further improve the previously performed learnings. Afterwards the "scr_status," the "loc_rcvr_status," and the "rem_rcvr_status" are validated. If all are positive, the link setup has been successful then, both PHYs will go into data mode SEND_N. If any of the status values is negative, the process restarts.

Table 4.3 gives an overview on the start-up sequence that has been defined for 1000BASE-T. 100BASE-T1 uses the same start-up procedure, except that all transmitted symbols can only have the values [−1 V, 0, +1 V] (see also Figure 4.30). Note that during start-up, the system runs a timer that defines the maximum time the system can remain in slave silent and training (idle) state. If expired before the loc_rcvr_status is OK, data transmission will not be enabled and the system will not change into SEND_N.

4.3.1.2 100BASE-T1/OABR

In autumn 2007, when BMW approached Broadcom and other Ethernet PHY semiconductor vendors in search for a 100 Mbps Ethernet technology suitable for use over unshielded cables in the harsh automotive environments, 1000BASE-T was long established and the 10 Gbps technology 10GBASE-T was just being introduced into the market. For optical systems, the standardization efforts for the 40 and 100 Gbps PHY standards were on the way [51]. With this in mind 100 Mbps seems a very low data rate. But, it was the experiences and building blocks that had been generated for the ever higher data rate PHYs that helped finding a solution for the automotive requirements at 100 Mbps.

Figure 4.22 shows the dependency of the attenuation (loss) as well as NEXT and FEXT interference on the frequency for the example 50 m Cat 6 channel. As can be

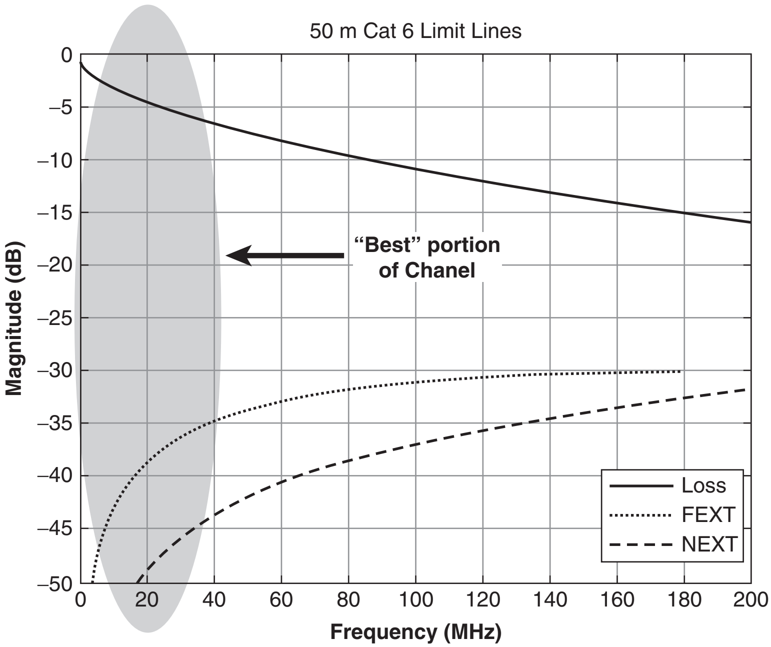

Figure 4.22 Loss (attenuation), NEXT, and FEXT as a function of frequency [52]. Figure reprinted with permission from Broadcom Corp., © 2012 Broadcom Corporation.

seen, the attenuation as well as the interference are smallest below 40 MHz. The bandwidth (Nyquist frequency) of both 100BASE-TX and 1000BASE-T was 62.5 MHz (see also Table 4.10). One of the key design points for BroadR-Reach was to reduce this bandwidth further, which helped with both, the longer channel it was intended for and the more stringent EMC requirements of the automotive industry. The development of 1000BASE-T had shown that a system can be designed to fulfill the same bandwidth requirements as its counterpart with 10 times less data rate, 100BASE-TX. With that perspective, about halving the bandwidth for a system with the same data rate as 100BASE-TX, as has been done for BroadR-Reach, seemed realistic.

Figure 4.23 shows the power spectral density for 100BASE-TX, 1000BASE-T, and BroadR-Reach/100BASE-T1. As can be seen, 1000BASE-T and 100BASE-TX both use about 125 MHz and therefore have a Nyquist frequency of 62,5 MHz. At a Baudrate of 66⅔ MHz, BroadR-Reach achieves a Nyquist frequency of 33⅓ MHz. The following technical description explains in details the functioning of the BroadR-Reach/OABR/100BASE-T1 technology, which made this result possible. Aggressive in-band filtering and a maximum cable length of 15 m are key elements.

The three main blocks of a 100BASE-T1 PHY are the PCS, the PMA, and management. The latter holds a serial interface (Management Data Input/Output (MDIO) and Management Data Clock (MDC)) that allows to write and read PHY register data. The standard [22] further segments the PCS into the three blocks "PCS transmit enable,"

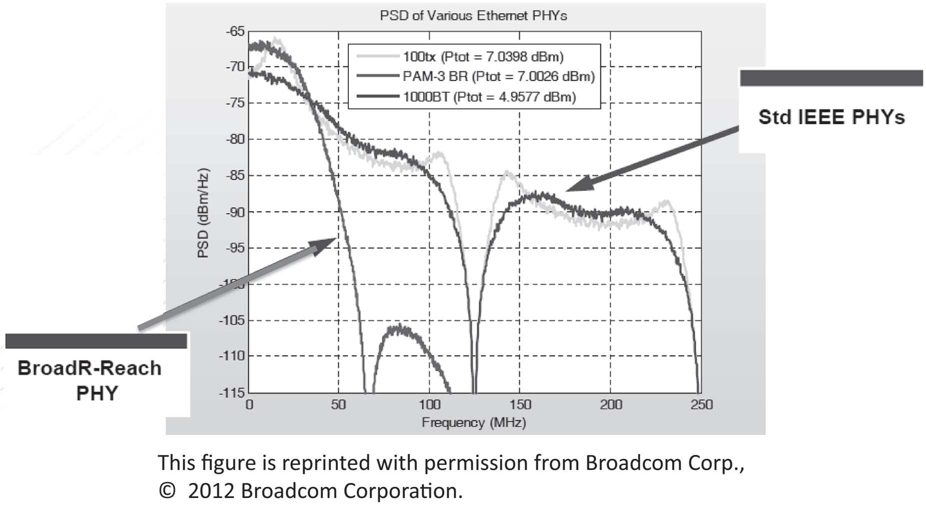

Figure 4.23 Power spectral density of 100BASE-TX, 1000BASE-T, and BroadR-Reach/100BASE-T1 transmit signals [52]. Figure reprinted with permission from Broadcom Corp., © 2012 Broadcom Corporation.

"PCS transmit," and "PCS receive" (see also Figure 4.24). The PMA consists of "PMA transmit," "PMA receive" and "clock recovery," which are all described in more detail below. Figure 4.25 provides an overview of the signaling in the PCS transmitter including the transmit enabling. The description of the PCS in the text follows the numbering provided in Figure 4.25 (for the PCS transmitter) and Figure 4.29 (for the PCS receiver).

1 **PCS transmit enable:** The PCS transmit enable converts the TX_ER and TX_EN signals into tx_error_mii and tx_enable_mii. If the link_status is not OK or the PCS is reset, then tx_enable_mii and tx_error_mii are FALSE. The situation changes when the tx_mode is in data transmission mode (SEND_N, see Section 4.3.1.1, Table 4.3). Then tx_enable_mii equals TX_EN and tx_error_mii equals TX_ER, else they remain "FALSE."
2 **Aligner:** The aligner adapts the lengths of tx_error_mii and tx_enable_mii such that they are fitted for PHY internal use. Adaptations of the duration of the values might be needed in case SSD/ESD/IDLE symbols are added to the data stream, or when stuff bits are used in the 4B3B conversion (see next bullet point and Figure 4.26). The input data tx_error_mii and tx_enable_mii are changed into the output data tx_error_n and tx_enable_n.
3 **4B3B conversion (bit reformatter):** This block regroups four incoming bits TxD[3:0] into groups of three tx_data[2:0]. To keep the same data rate of 100 Mbps the clock rate needs to change for tx_data[2:0] from 100 Mbps = 4 × 25 Mbps to 3 × (25 Mbps∗4/3) = 3 × 33⅓ Mbps = 100 Mbps. When the incoming data is not a multiple of three, one or two stuff bits have to be inserted (see Table 4.4 for an example). The value of the stuff bits is not specified in the standard. However, their value

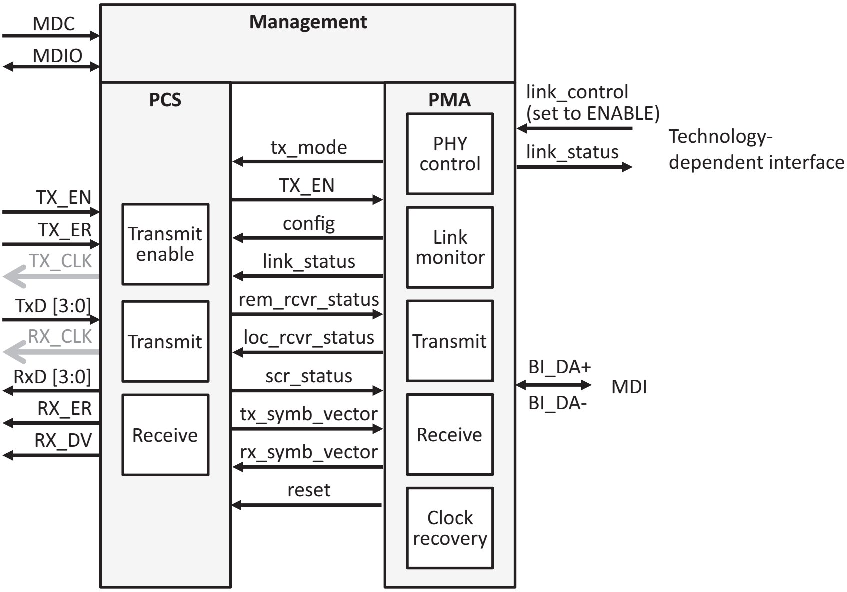

Figure 4.24 Building blocks of a 100BASE-T1 PHY [22]. Normally the link_control is enabled with the completion of the autonegotiation. As 100BASE-T1 does not support autonegotiation, the value is set to enable with power on.

is not important as they are simply removed again on the receiver side. In case stuff bits are added at the end of the data portion of the Ethernet packet, the aligner has to adapt the lengths of the tx_error_mii and tx_enable_mii accordingly (see Figure 4.26 for examples).

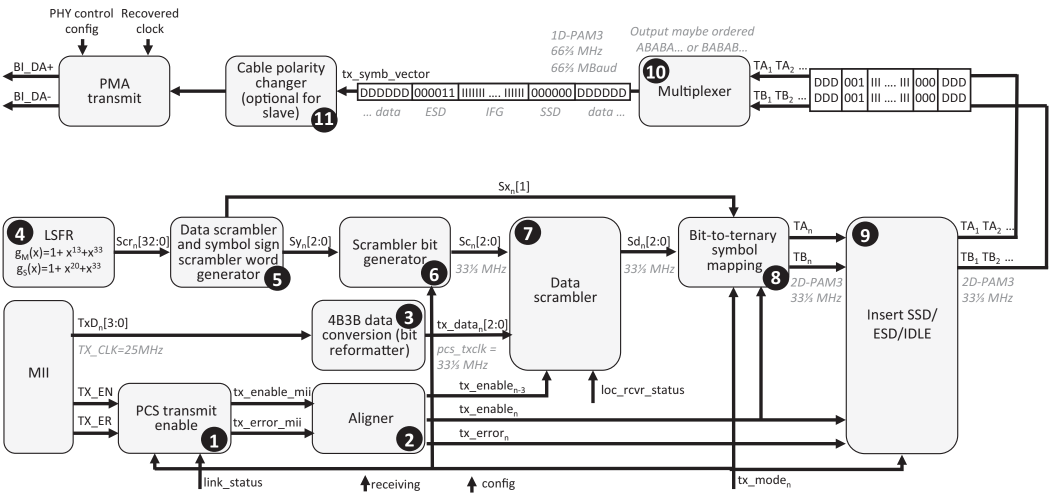

Figure 4.25 Overview of the PCS transmitter and transmit enable functions for 100BASE-T1.

Table 4.4 Example 4B3B conversion of two-byte data requiring two stuff bits (given the value "0" in the example)

25 MHz			TxD_3				TxD_2				TxD_1				TxD_0			
TxD[x]			3	2	1	0	3	2	1	0	3	2	1	0	3	2	1	0
data			1	1	1	0	0	1	1	0	1	0	1	1	0	0	0	1
data stuffed	0	0	1	1	1	0	0	1	1	0	1	0	1	1	0	0	0	1
tx_data [x]	2	1	0	2	1	0	2	1	0	2	1	0	2	1	0	2	1	0
33⅓ MHz	tx_data_5			tx_data_4			tx_data_3			tx_data_2			tx_data_1			tx_data_0		

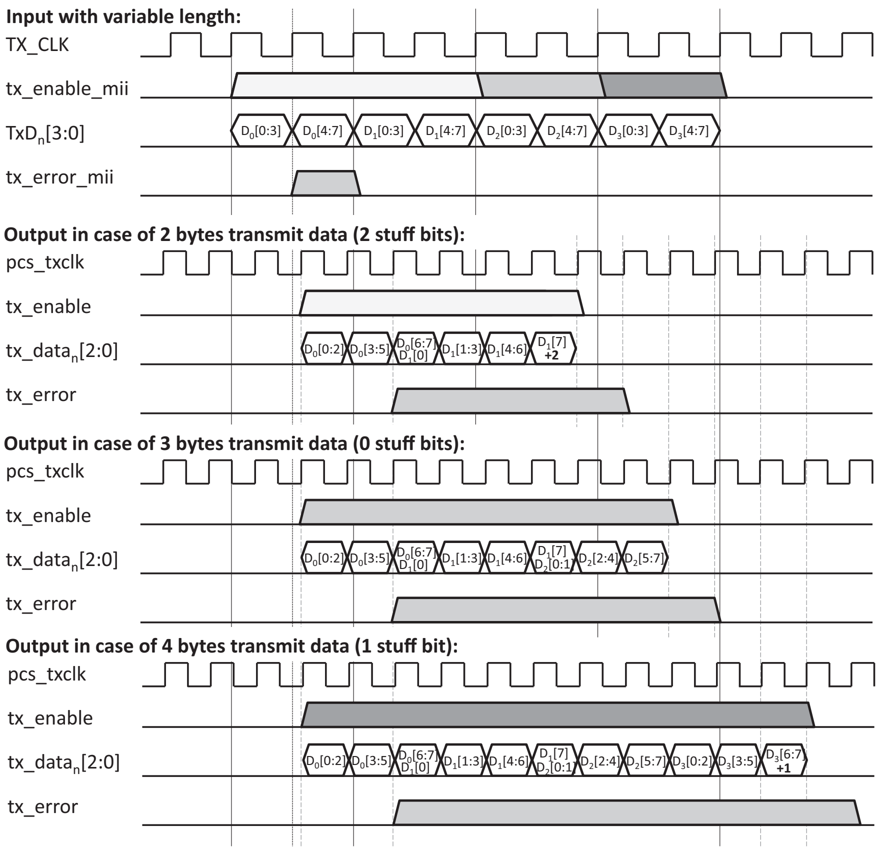

Figure 4.26 4B3B bit mapping and aligner extension of tx_enable and tx_error in case of different number of bytes $D_n[0:7]$ at the end of the packet [22]. In the example depicted, tx_error is set TRUE (= "1") to show how it is handled. Naturally, being able to use the data in the receiver requires tx_error to be FALSE (= "0").

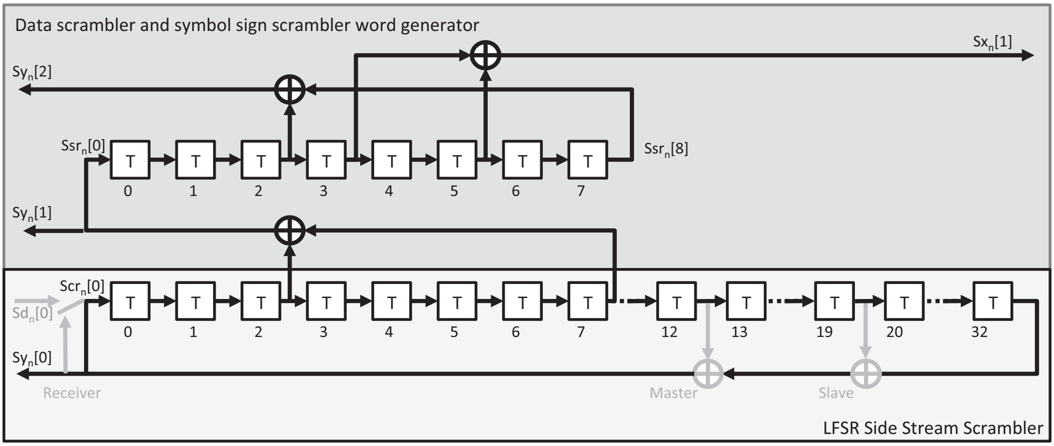

Figure 4.27 LFSR and data scrambler symbol sign word generator for 100BASE-T1/OABR (for an explanation of the receiver change, see point 13).

4 **Linear Feedback Shift Register (LFSR):** The LFSR has the purpose of creating the pseudo-random binary starting sequence used for randomizing/scrambling the transmit data. The 33 initial bits the LFSR holds are not specified, but are up to the implementer. These initial bits can have any of the 2^{33} possible values, except being all "0." In that case the created output sequence would always stay "0" – 0 exor 0 equals 0 – and defy its purpose of being pseudo-random. For any other starting value the resulting output sequence is repeated after $2^{33} - 1$ register shifts. Note that this is not necessarily the case for every LFSR with the length 33, but due to the specific polynomials chosen.[7] With a frequency of 33⅓ MHz and thus 30 ns register shift, the sequence repeats itself every $(2^{33} - 1) \times 30$ ns = 257.69 s. In order to be able to reuse as much as possible from 1000BASE-T, 100BASE-T1 uses the same master and slave polynomials as 1000BASE-T (see Equation 4.4 and Figure 4.27). Both polynomials are always needed. In the master device, the master polynomial for the transmit data and the slave polynomial for the receive data and in the slave vice versa. Note that the scrambler for 1000BASE-T1 (see Section 4.3.2.1) has been further optimized and is shorter.

5 **Data scrambler and symbol sign scrambler word generator:** Figure 4.27 shows how the starting pseudo-bit stream is generated in the LFSR and transformed into pseudo-random, three bit words $Sy_n[2:0]$ by the symbol sign scrambler word generator. The value $Sx_n[1]$ is used for randomizing the IDLE symbols during tx_mode = SEND_ N (see also point 8 for bit to ternary mapping). This is a reduced version of how it is done for 1000BASE-T. Note that because 100BASE-T1 uses only one of the four $Sx_n[3:0]$ values provided in 1000BASE-T, the 100BASE-T1 specification refers to it as Sx_n only, without numbering the bit [1]. This book keeps the $Sx_n[1]$ nomination in order to show the derivation from the 1000BASE-T standard. The second shift register shown in Figure 4.27 that depicts the function of the symbol sign word generator can be circumvented by applying Equation 4.5 directly to the

LSFR output.

$$\begin{aligned}
Sy_n\,[0] &= Scr_n\,[0] \\
Sy_n\,[1] &= Scr_n\,[3] \oplus Scr_n\,[8] \\
Sy_n\,[2] &= Scr_n\,[6] \oplus Scr_n\,[16] \\
Sx_n\,[1] &= Scr_n\,[7] \oplus Scr_n\,[9] \oplus Scr_n\,[12] \oplus Scr_n\,[14]
\end{aligned} \tag{4.5}$$

6 **Scrambler bit generator:** The scrambler bit generator transforms or keeps the data depending on the tx_mode value. If the tx_mode is SEND_Z the output is all "0." In all other cases the output equals the input.

7 **Data scrambler:** In case the $tx\_enable_{n-3} = 1$, the data scrambler finally performs the bit by bit exor combination between the transmit data and the pseudo-random sequence derived as described above $Sd_n[2{:}0] = Sc_n[2{:}0] \oplus tx\_data\,[2{:}0]$. The exor function achieves that the pseudo-random/scrambler sequence inverses the transmit data, if the bit in the pseudo-random sequence is a "1" and keeps the transmit data as is, if it is a "0." Because the scrambling bits use the same frequency as the transmit data the bandwidth/data rate of the resulting sequence $Sd_n[2{:}0]$ stays the same as for $Sc_n[2{:}0]$ and tx_data [2:0]. The transmit power is therefore not spread over a larger frequency range by the operation but simply more evened out within the same frequency band. It also means that the DC portion of the signal is reduced, which is advantageous for the two capacitors in the transmission line. Electromagnetic interference is therefore reduced. In case $tx\_enable_{n-3} = 0$, the first two bits of each data triplet stay unchanged $Sd_n[1{:}0] = Sc_n[1{:}0]$, while the third bit, $Sc_n[2]$, is either inverted if loc_rcvr_status = OK ($Sd_n[2] = !Sc_n[2]$), i.e., receiver part of the PHY operates correctly, or stays the same if the loc_rcvr_status = not OK ($Sd_n[2] = Sc_n[2]$). This allows the communication partner to detect, whether it can set its rem_rcvr_status also to OK or not OK. The latter is relevant in the start-up phase, shown in Table 4.3 and Figure 4.30. Another important parameter during start-up is the scr_status, which indicates whether the scrambler in the receive path has synchronized or not.

To start with, all local_rcvr_status, rem_recv_status, and scr_status values are not OK. The master first sends IDLE symbols (SEND_I), while the slave is in SEND_Z, i.e., zero voltage is being put on the line. The slave uses the bit $Sd_n[0]$ of the received idle symbols to synchronize its master descrambler. When the descrambler is synchronized, the scr_status in the slave is set OK. Once this happened, the slave will start sending IDLE symbols, too, and can recognize rem_rcvr_status of the master encoded in the IDLE symbols. As explained, the information is transmitted with $Sd_n[2]$, which is why the distinction is so important. The master can then start synchronizing its own scrambler. When the receiver is the master is ready (scr_status = OK represents only one element or loc_rcvr_status = OK) the transmitted $Sd_n[2]$ bit will be changed accordingly, in order to indicate to the slave the changed status (received as rem_rcvr_status OK). For more details see also point 13 in the receiver section below.

Table 4.5 3B2T mapping used in 100BASE-T1 during normal transmission (tx_mode = SEND_N, and tx_enable = 1) in contrast to a respective Gray code as used for 1000BASE-T1 [22] [12]

100BASE-T1 $T_B\rightarrow$ $T_A\downarrow$	−1	0	1
1	101	110	111
0	011	SSD/ESD	100
−1	000	001	010

1000BASE-T1 $T_1\rightarrow$ $T_0\downarrow$	−1	0	1
1	011	111	110
0	010	SSD/ESD	100
−1	000	001	101

8 **Bit to ternary symbol mapping (3B2T):** This block translates the three bits $Sd_n[2:0]$ onto the ternary transmit symbols TA_n and TB_n used for PAM3. The ternary values of TA_n and TB_n [−1, 0, 1] are mapped directly onto the voltage levels [−1, 0, 1]V. The used 3B2T code does not guarantee that the signal is DC-free, nor that the clock is continuously available, nor does it use a Gray code. A Gray code normally ensures that successive code words only differ by 1 bit in order to limit the errors in case of continuously changing signals [53] an thus reduces the Bit Error Rate (BER).[8] Table 4.5 shows the 100BASE-T1 3B2T-mapping during data transmission SEND_N in contrast to an example Gray code like used for 1000BASE-T1. The mapping $\{TA_n, TB_n\} = \{0, 0\}$, is not used for data, but reserved for the Start Stream Delimiter (SSD) and parts of the End Stream Delimiter (ESD). The mapping requires that always three bits are available. This is why, should the user data not divide by three, the data is extended by stuff bits, as explained above.

The exact mapping depends onto a number of parameters, first of all on the tx_mode. In case tx_mode = SEND_Z, TA_n and TB_n are mapped to 0, which, in the end, is the meaning of SEND_Z. In case of the idle mode tx_mode = SEND_I the data is mapped according to Table 4.6. In this case neither values tx_enable or $Sx_n[1]$ are taken into account and the scrambling might not be as effective. This potentially results in a slightly worse EMC performance. However, as explained above, SEND_I represents the training mode during which the scrambler and other aspects of the receiver are adjusted and synchronized. Only after this has been successful, the receiver can actually generate and use the value $Sx_n[1]$ out of its own descrambler. To counterbalance this and improve the robustness against EMC impacts, only six out of the nine possible $\{TA_n, TB_n\}$ combinations are used during training SEND_I. The specific selection allows to distribute the transmit power better across the used frequency band.

This is similar, when the system is idle in case of tx_mode = SEND_N but tx_enable = 0 (i.e., link acquisition has been completed, but the transmitter MII has no data to send and is therefore also in an idle state). Here the incoming bits are also mapped to six $\{TA_n, TB_n\}$ combinations only, while the scrambling bit $Sx_n[1]$ now additionally decides on the exact values, in order to balance the power density better.

Finally, during normally transmission when tx_mode = SEND_N and tx_enable = 1, eight of the nine possible ternary symbol combinations are used for $\{TA_n, TB_n\}$

Table 4.6 Bit to ternary symbol mapping for 100BASE-T1

	tx_mode = SEND_Z		tx_mode = SEND_I or tx_mode = SEND_N, tx_enable = 0, $Sx_n[1] = 0$		tx_mode = SEND_N, tx_enable = 0, $Sx_n[1] = 1$		tx_mode = SEND_N, tx_enable = 1	
$Sd_n[2:0]$	TA_n	TB_n	TA_n	TB_n	TA_n	TB_n	TA_n	TB_n
000	0	0	−1	0	−1	0	−1	−1
001	0	0	0	1	1	1	−1	0
011	0	0					0	−1
010	0	0	−1	1	−1	1	−1	1
SSD/ESD	0	0	Not used {00}, {11}, {−1−1}		Not used {00}, {01}, {0−1}		0	0
100	0	0	1	0	1	0	0	1
101	0	0	0	−1	−1	−1	1	−1
111	0	0					1	1
110	0	0	1	−1	1	−1	1	0
Mode	"inactivity"		"training" and/or "idle"		"idle"		"data"	

(see also Table 4.6). The combination {00} is used for control purposes, i.e., before and after the actual transmit data, the SSD and ESD encoded in the {00} indicate to the receiver the beginning and end of transmit data (see point 9).

9 **Insert SSD/ESD/IDLE:** Depending on the values of tx_enable and tx_error, this block inserts control symbols into the data stream. With the end of the idle state and data ready to transmit, the value tx_enable changes from FALSE = 0 to TRUE = 1. This information comes from the MII interface where TX_EN changes when data is available for transmission. When tx_enable has changed, first of all three SSD PAM3 pairs [00 00 00] are inserted into the data stream. If tx_enable is still TRUE after their completion, the user data is inserted. As soon as tx_enable changes to FALSE the ESD is added. If an error was detected in the packet (the information comes with TX_ER from the MII) and tx_error is TRUE = 1 then the ESD information is [00 00 −1 −1]. The PHY forwards the packet regardless. The layer behind the MII, e.g., the switch, might decide to drop it. If tx_error is FALSE, i.e., no error has been detected before sending, the ESD is [00 00 11]. Figure 4.28 depicts the functioning.

The insertion of additional symbols with SSD and ESD adds additional symbols to the data stream that the MII is not aware of. In order not to desynchronize the transmission or to change data rate between MII and data on the channel, the following method is applied. Each Ethernet packet normally starts with a seven byte preamble (see Section 1.2.1, Figure 1.5). In order to accommodate for the SSD, the 100BASE-T1 (and also the 1000BASE-T1) PHYs shortens this preamble. The SSD comprises six 1D-PAM3 symbols or three 2D-PAM3 symbols respectively. Because

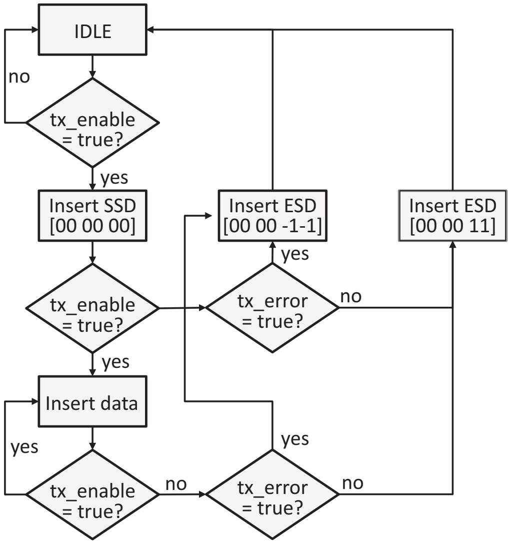

Figure 4.28 Insertion of control symbols in the "Insert SSD/ESD/IDLE" block. This happens only during SEND_N status; tx_enable defines whether the status is nevertheless idle.

three bits are mapped to two PAM3 symbols in the 3B2T conversion, the six PAM3 symbols represent $6*3/2 = 3*3 = 9$ bits. Thus, the ternary symbols that represent the scrambled nine bits of the original preamble [101 010 101] are replaced with the SSD symbols [00 00 00]. This is possible because in the switched Ethernet network, the original function of the preamble is no longer needed. The preamble was originally needed to allow synchronization at the beginning of every packet in a large network with bus topology (potentially including hubs) that did not have a continuous connection. With the introduction of the switched architecture, the preamble, and also the length of the InterFrame Gap (IFG) have been kept in the standard for backward compatibility reasons only, but without functional necessity. In consequence, the preamble is shortened for the SSD and potential stuff bits and the ESD shorten the IFG. The receiver has to ensure that the original timing with preamble and IFG is reinstalled (and SSD and ESD are removed), before passing the data on to the MII. Table 4.9 (see point 15) explains the principle realignment between the MII data and the transmit data.

10 **Multiplexing 2D-PAM3 to 1D-PAM3:** This block multiplexes the two parallel ternary symbols TA_n and TB_n into a one-dimensional data stream, before the PMA transmit block processes the data into BI_DA+ and BI_DA− and puts them onto the channel. With the multiplexer the symbol rate doubles from 33⅓ MBaud to 66⅔ MBaud and the symbol duration halves from 30 ns to 15 ns. It is not specified whether the output of the multiplexer starts with TA_n or TB_n. The receiver thus has to comprise a function, which can recognize the order of the symbols and acts on it (see point 14).

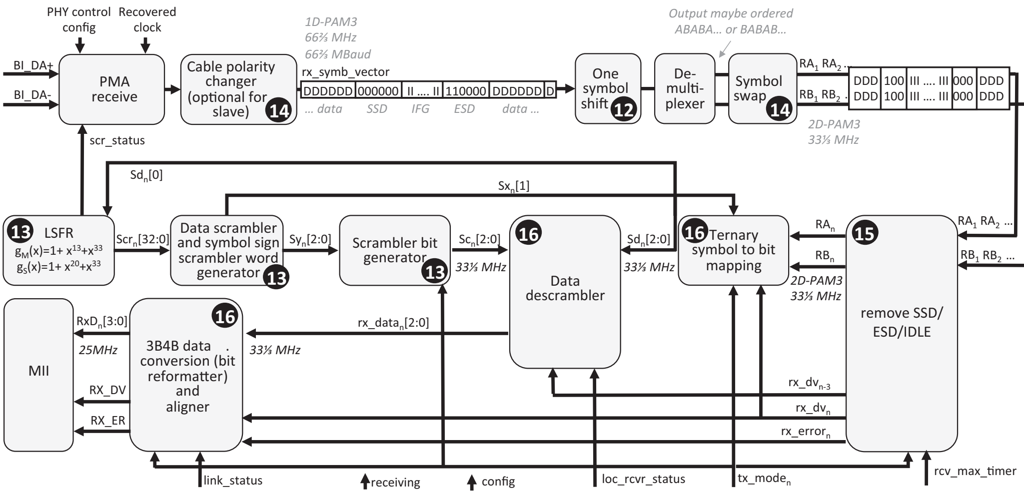

Figure 4.29 Example for elements of a 100BASE-T1/OABR PCS receiver.

11 **Optional cable polarity changer:** The 100BASE-T1 specification includes an optional cable polarity changer for the slave. The slave can detect a polarity change for the first time during link acquisition in its receive path, once it has completed its descrambler synchronization (see also point 14). After the master's idle data has been recognized, the cable polarity changer can detect a polarity change and correct it, if necessary. If this is indeed necessary, the slave also has to change the polarity of its transmit symbols, as the specification does not foresee a polarity change in the master. A polarity change means $1 \rightarrow -1$, $0 \rightarrow 0$, and $-1 \rightarrow 1$.

The standard specifies significantly fewer parts of the receiver than of the transmitter. To assure interoperability and compliance, the receiver has to be able to count on specific properties of the received signal, but how it handles them is a part of the Unique Selling Point (USP) of the implementer and thus generally not specified. The following therefore describe examples. In principle, the PCS receiver performs most operations of the PCS transmitter in reverse order as is depicted in Figure 4.29. The different blocks are described in more detail below.

12 **One-symbol shift:** In order to receive data correctly, the PCS receiver has to group the received symbols RA_n, RB_n into pairs with correct polarity, correct order (RA_n RB_n or RB_n RA_n) and same timing RA_n and RB_n. Otherwise the ternary symbol to bit mapping will produce the wrong output. The receive PCS will also need to synchronize its scrambler, so that it can descramble the transmission data correctly. To start with the receiver has no information, but has to detect all that is necessary from the data it receives during training mode.

The very first thing, before it makes sense to start the process of scrambler synchronization, the symbol grouping needs to be correct and potentially corrected. For

Table 4.7 Possible grouping of 2D-PAM3 symbols during training

Received symbol stream	RA_{n-1}	RB_{n-1}	RA_n	RB_n	RA_{n+1}	RB_{n+1}
Correct grouping	RAB_{n-1}		RAB_n		RAB_{n+1}	
Incorrect grouping		$RAB_{n-1,\,n}$		$RAB_{n,\,n+1}$		

or

RB_{n-1}	RA_{n-1}	RB_n	RA_n	RB_{n+1}	RA_{n+1}
RBA_{n-1}		RBA_n		RBA_{n+1}	
	$RBA_{n-1,\,n}$		$RBA_{n,\,n+1}$		

the symbol grouping, the receiver has to detect, whether independent of the symbol order RA_n RB_n or RB_n RA_n, RA_n is grouped with RB_n and not accidently with RB_{n-1} or RB_{n+1} and vice versa (see Table 4.7). Basis for the detection of the status of the symbol grouping is the 3B2T coding as shown in Table 4.6 and Table 4.8. As can be seen in both tables, during SEND_I only six of the nine possible 2D-PAM3 symbols are used and only these symbols should be received. If the symbol grouping is wrong, the 2D-PAM3 symbols {00}, {11}, {−1−1} will be received also. In this case, the symbol grouping needs to be shifted by one symbol.

13 **(De-)Scrambling:** As has been explained above, the PCS receiver first has to synchronize its scrambler, before it can sensibly receive any data. The scrambler polynomials used are the same as defined for the transmitter LFSR, only that in the receive path the slave will use the master polynomial for reception and the master will use the slave polynomial. Additionally, in the receiver path the scrambler holds

Table 4.8 Transmit and possible receive symbols and bits during training mode (tx_mode = SEND_I), depending on cable polarity and AB symbol order

Transmitted bits and symbols (see Table 4.6)					Received symbols and bits during SEND_I															
					Polarity OK AB order OK				Polarity NOK AB order OK				Polarity OK AB order NOK				Polarity NOK AB order NOK			
$Sd_n[x]$			TX_n		RX_n		$Sd_n[x]$		RX_n		$Sd_n[x]$		RX_n		$Sd_n[x]$		RX_n		$Sd_n[x]$	
2	1	0	A	B	A	B	2	0	A	B	2	0	A	B	2	0	A	B	2	0
0	0	0	−1	0	−1	0	0	0	1	0	1	0	0	−1	1	1	0	1	0	1
0	x	1	0	1	0	1	0	1	0	−1	1	1	1	0	1	0	−1	0	0	0
0	1	0	−1	1	−1	1	0	0	1	−1	1	0	1	−1	1	0	−1	1	0	0
SSD/ESD (not used)			{00}{11} {−1−1}		Not used				Not used				Not used				Not used			
1	0	0	1	0	1	0	1	0	−1	0	0	0	0	1	0	1	0	−1	1	1
1	x	1	0	−1	0	−1	1	1	0	1	0	1	−1	0	0	0	1	0	1	0
1	1	0	1	−1	1	−1	1	0	−1	1	0	0	−1	1	0	0	1	−1	1	0
							3)	1)			3)	1)				2)				2)

a selector that can open or close the feedback loop in the LFSR (see Figure 4.27). To begin with, the feedback loop is open, making the LFSR a shift register without feedback.

When the receiver is activated into training mode, the PCS receiver will start filling its register with the data $Sd_n[0]$ it receives from the master and that has been decoded in the 2T3B ternary symbol to bit mapping (see also Figure 4.27). Ideally, it takes 33 shifts (33×30 ns $= 0.99$ μs) to fill the register. However, to start with, the PMA has to adjust properly (especially clock, equalizer, AGC) and the one-symbol shift has potentially to be performed, before $Sd_n[0]$ will represent usable values (for cable polarity and symbol order see below). Only then the LFSR has the chance to successfully compare the scrambler output $Sy_n[0] = Scr_n[0]$ with the incoming data $Sd_n[0]$ and to synchronize. Once $Scr_n[0] = Sd_n[0]$ are continuously, i.e., long enough, identical, synchronization can be assumed. The standard does not describe when the synchronization has been completed. It is up to the implementer to decide when to set the scr_status to OK and to close the feedback loop. Once the LFSR is synchronized, the scrambler values $Sy_n[2{:}0]$ and $Sx_n[1]$ can be generated for 2T3B mapping in normal mode SEND_N and consecutive scrambler operations.

14 **Cable polarity changer and symbol swap:** However, the scrambler will not lock, when the selected symbol order RA_n RB_n or RB_n RA_n is not the same as the transmitted order. It is described in the following, how this can be detected with help of the scrambling and that the correct cable polarity is not necessary to do so. The cable polarity can be corrected, once the symbol order is correct and the scrambler has locked (see below). In order for the scrambler to lock, it has to receive $Sd_n[0]$ correctly, even if the polarity is changed and/or the symbol order was swapped. To explain how this can be done, Table 4.8 shows what happens during training mode (tx_mode = SEND_I) in the 2T3B conversion in the different (error) cases.

In the left most columns in Table 4.8 shows the 3B2T mapping of the transmission bit triplets into the 2D-PAM3 symbols during training mode SEND_I as presented in Table 4.6. The following columns show what the receiver receives as RA_n, RB_n and consequently decodes as $Sd_n[0]$, $Sd_n[2]$ for the different scenarios. The first thing that can be noticed when comparing the columns marked 1) and 2) with each other is that $Sd_n[0]$ is actually not influenced by a polarity change, but only by a symbol order mismatch. In case of correct symbol order $Sd_n[0] = 1$ for $RA_n = \{1 \text{ or } -1\}$ and $Sd_n[0] = 0$ for $RA_n = \{0\}$, while it is the other way around for incorrect symbol order, i.e., $Sd_n[0] = 0$ for $RA_n = \{1 \text{ or } -1\}$ and $Sd_n[0] = 1$ for $RA_n = \{0\}$. This means that even with wrong cable polarity, the scrambler can lock, as long as the symbol order is right. When starting, the slave can simply assume a certain symbol order, and if the scrambler does not synchronize within a certain time, it performs a symbol swap and restarts the synchronization effort of the scrambler.

With the scrambler synchronized the correct values $Sy_n[2{:}0]$ and $Sc_n[2{:}0]$ can be generated within the receiver. In training mode, the generated $Sc_n[2{:}0]$ should be the same as the received $Sd_n[2{:}0]$. This can be used to detect a polarity change. As shown in the columns marked 3) in, $Sd_n[2]$ is affected by a polarity change

Table 4.9 RA_n and RB_n mapping onto bits

	. . . data					SSD			Idle					ESD			data . . .		
RA	D	D	D	. . .	D	0	0	0	I	I	. . .	I	I	1	0	0	D	D	D
RB	D	D	D	. . .	D	0	0	0	I	I	. . .	I	I	1	0	0	D	D	D
rx_data[2]	d	d	1	. . .	0	1	0	1	0	0	. . .	0	0	0	0	0	d	d	d
rx_data[1]	d	d	1	. . .	1	0	1	0	0	0	. . .	0	0	0	0	0	d	d	d
rx_data[0]	d	d	0	. . .	0	1	0	1	0	0	. . .	0	0	0	0	0	d	d	d
	. . . data		SFD and preamble						IFG								data . . .		

and will result in exactly the reverse value from what it should be if the polarity assumed is wrong. Once such a behavior has been detected, the polarity can be changed.

Note, that $Sd_n[2]$ is also used to indicate to the receiver the rem_rcvr_status (see also point 7) and that it might potentially intentionally be inverted by the transmitter. Using it to discover a polarity change is nevertheless possible, because of the order of events. During start-up, the loc_rcvr_status in the master will never change into OK and the bit $Sd_n[2]$ will not be inverted, before the slave has not started transmitting SEND_I idle symbols (with the right cable polarity). The slave will not start transmitting idle symbols before its scrambler has not locked. However, as soon as it scrambler is locked, it can detect and correct a necessary cable polarity change. It will thus long have terminated its use of the $Sd_n[2]$ values for the cable polarity detection, before the master might invert them to indicate his loc_rcvr_status as OK.

15 **Remove SSD/ESD/IDLE:** In normal transmission mode SEND_N and tx_enable = 1, the transmitter replaced parts of the preamble and the IFG with the SSD and ESD (see point 9). The receiver has to reverse this. For the SSD, the block has to reinsert the PAM3 symbols that represent the preamble and that, following 2T3B decoding and descrambling result in the 9 bits [101 010 101]. For the ESD it has to be ensured that the state of the IFG is reinstalled (see Table 4.9 for the principle functioning) and that its values are replace by "0."

In respect to the ESD, the standard foresees a measure to ensure that the receiver does not unintentionally stay in data mode, because it misses the detection of the first two 2D-PAM3 symbols of the ESD [00 00]. The standard therefore introduces state machine that uses the value rcvr_max_timer. If a timer with rcvr_max_timer is expired before the change into idle state was initiated otherwise, the idle state in the PCS receiver is enforced.

16 **Data recovery and alignment:** The other activities in the PCS receiver are performed in reverse order. First, in the following 2T3B decoding, the rules as described in Table 4.6 are applied to obtain the values $Sd_n[2:0]$. Provided, tx_mode = SEND_N, these are then scrambled with the correct scrambling triplets $Sc_n[2:0]$ that reverse the previously applied randomization by inversion if $Sc_n[x] = 1$ and

keeping the data as is if $Sc_n[x] = 0$; just like it was done in the transmitter. The 3B4B conversion groups and reschedules the data from groups of three bits at 33⅓ MHz to groups of four bits at 25 MHz. The number of bits after 2T3B decoding needs to be a multiple of four. If there are additional bits left, these are dropped as stuff bits and not used for the 3B4B bit mapping (reverse operation as shown in Table 4.4). The aligner adapts the lengths of rx_dv and rx_error, to forward them as RX_ER and RX_DV to the MII interface.

The PMA part of the Ethernet transceiver consists of the PHY control, the link monitor, the transmit and receive functions as well as the clock recovery (see Figure 4.24). The 100BASE-T1 PMA transmit and receive functions are very similar to what has been explained with 1000BASE-T in Section 4.3.1.1 and Figure 4.21. The following description therefore focusses on the essentials.

In the **PMA transmit**, the signals TA_n and TB_n coming from the PCS transmitter are first of all converted into analog signals. Other than for 1000BASE-T, no additional partial response operation is performed on the signals for 100BASE-T1 before the D/A. Following the D/A an internal filter can be used in order to ensure that the output signal meets the PSD mask defined in the specification. The hybrid then couples/decouples the transmit and receive signals onto one channel. An additional low pass filter can be used on the signal following the hybrid to improve its EME and EMI behavior. Such filter would need to be effective from 33⅓ MHz on and can be PHY internal or PHY external.

The **PMA receive** has the task to detect the information that arrives and to correctly convert it into PAM3 symbols that are then passed on with RA_n and RB_n to the PCS receiver. Because of the baseband transmission used with 100BASE-T1, the signals typically arrive at the receiver end of the transmission attenuated (smaller amplitude) and distorted (changed in shape) [54]. High and low pass filters, as well as amplitude correction (for AGC see description for 1000BASE-T in Section 4.3.1.1) can help to restore the received waveforms. However, for receiving data correctly it is extremely important that the clock runs at the right frequency and phase, in order to be able to sample the incoming waveform at the right time.

The PMA therefore also needs to perform the **clock recovery**. The reference clock of 66⅔ MHz is set by the link partner that is master. The master obtains its clock value from a local clock source, making it available within its own system as well as imposing it onto the transmit signal. The slave then recovers frequency and phase from the incoming signal, and having done so, in return imposes the recovered clock onto its own transmit signal. The master knows the frequency the received signals have, because of its own clock source. But, it still has to extract the correct phase from the incoming signals on its side, thus completing the loop-timing applied. The loop-timing concept is the same as for 1000BASE-T (and in 1000BASE-T1). Note that because of the long scrambler sequence, baseline wander is not an issue for 1000BASE-T or 100BASE-T1 and compensational measures are not necessary (other than for 100BASE-TX).

The PMA receiver then also performs the A/D conversion and echo cancellation that is needed because reflections of the own master transmit signal on the transmission channel can distort the received signal. Like it has also been described for 1000BASE-T, the receiver will perform some form of equalization (DFE, FFE) before the demodulator decides on the PAM3 signal values of RA_n and RB_n to be forwarded to the PCS. Because there is no convolutional encoding like for 1000BASE-T, 100BASE-T1 does not require a Trellis decoder for this. For 100BASE-T1 a simple slicer is sufficient for this task. Once the receiver perceives that it is able to do so correctly as well as continuously and the PCS has announced that its scrambler locked with scr_status = OK, the unit will convey this with the loc_rcvr_status = OK to the rest of the system and the connected unit.

In contrast to, e.g., a mobile communication channel, the changes the 15 m maximum wireline channel of Automotive Ethernet experiences are harmless. Nevertheless, it is advisable that the filter coefficients of a potential FFE and DFE can be adjusted dynamically. This ensures optimal reception in case the channel changes, e.g., because of heat, humidity, aging, or mechanical strain. In case of burst errors, 100BASE-T1 relies on a relatively high SNR. For systems like 1000BASE-T1 (see Section 4.3.2.1) that work with a significantly lower margin, an additional Forward Error Correction (FEC) is needed to improve the correct reception of the data.

The **PHY control** determines the procedure that enables the units to exchange data (and with it the tx_mode the transceiver is in, SEND_Z, SEND_I, or SEND_N) and initiates the changes from one tx_mode to the next. The main parameters used to initiate changes are scr_status, loc_rcvr_status, rem_rcvr_status, as well as two timers maxwait_timer and minwait_timer. Most of the changes are initiated during the link-acquisition procedure, which has been explained in Section 4.3.1.1. While Table 4.3 in Section 4.3.1.1 shows the interrelation of the behavior of both units in sequence, Figure 4.30 shows the events for each units individually, including the use of the timers.

Figure 4.30 shows that the main difference in behavior between master and slave is at the beginning. When a unit that has been assigned master it simply goes directly into SEND_I state, whereas the slave does so only when its scrambler has locked and the scr_status is OK. This means that the slave will remain in SEND_Z somewhat longer than the master, while the master can be expected to remain longer in SEND_I (which can be derived from Table 4.3 but not from Figure 4.30). Each unit remains in training/SEND_I maximum for the time it takes to complete the training and set loc_rcvr_status to OK or for the time it takes for the maxwait_timer to expire; whichever is shorter. Should the maxwait_timer (200 ms ± 1% for 100BASE-T1)[9] expire before the loc_rcvr_status is OK, the link_status will go into not OK and the system will decide on a higher layer to restart the process and to go back to SEND_Z (hence the dotted line in Figure 4.30). When the unit, master or slave, has successfully reached loc_rcvr_status OK, they will check the status of the rem_rcv_status, i.e., whether the other unit they are communicating with is ready to receive, too. If the rem_rcvr_status is OK, data transmission mode SEND_N can begin. If the rem_rcvr_status is not OK, the system will remain in SEND_I to give the other unit more time to complete training (even though its own training was completed). If the rem_rcvr_status changes into OK, data transmission SEND_N will begin.

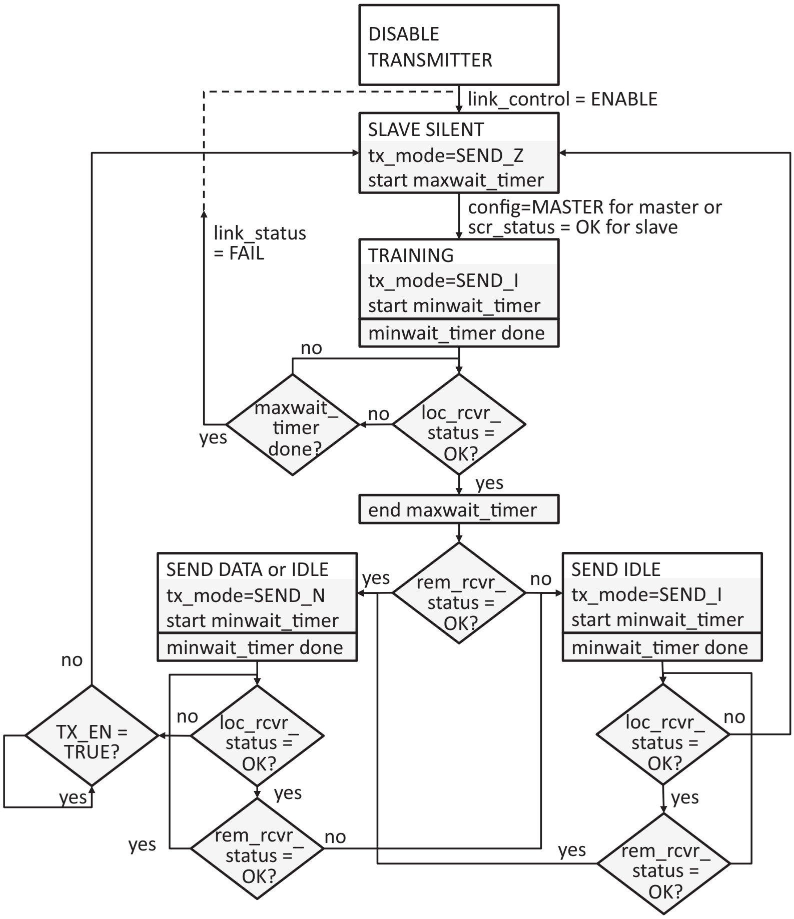

Figure 4.30 Master and slave PHY control sequence diagrams.

Data transmission SEND_N will continue for as long as there is data to transmit (TX_EN = TRUE) and the loc_rcvr_status and rem_rcvr_status are OK. If at any time the loc_rcvr_status changes from OK to not OK, e.g., because of a sudden burst of noise, the system will go back into SEND_Z once the unit has finished transmitting the last packet and TX_EN goes into FALSE. If the rem_rcvr_status changes into not OK, the transmission will go from SEND_N to SEND_I in order to allow for the rem_rcvr_status to recover. Whenever the unit changes into a new mode, the units run the minwait_timer, which determines the minimum time each unit has to stay in each new mode (1.8 μs $\pm$ 10% for 100BASE-T1). This is added to ensure that the different states do not change too fast, as this could destabilize the whole system.

The **link monitor** function has the purpose to support the PHY control by determining the status of the underlying channel and by communicating the status with the link_status parameter. The state link_status = FAIL = link down can have different reasons. For one link_status goes into fail, if PMA is reset or if the link_control is not enabled.[10] Furthermore, as has been explained with the PHY control above, the link_status goes into FAIL if the receiver has not been able to synchronize before the maxwait_timer expired. Should the link achieve synchronization and the loc_rcvr_status is OK during link_status = FAIL, the link monitor starts a stabilize timer. If the loc_rcvr_status is still OK when the time has expired, the link_status parameter will be set to OK and remain in this state. It will only leave this state, when the loc_rcvr_status changes into not OK and the link is not in training mode. In order for the link_status not to interfere with the training mode, a link_status OK can change into link_status not OK only when the maxwait_timer is not running (it does run during training).

Fast start-up is an important requirement for in-vehicle networking technologies. The 100BASE-T1 standardization project thus included the objective to achieve a valid transmission and receiving state from power on within less than 100 ms. The 100BASE-T1 specification thus explicitly specifies in the link monitor section that the time from power_on = TRUE to link_status = OK shall be smaller than 100 ms.

Table 4.10 compares different parameters for the different PHY technologies 100BASE-TX, 1000BASE-T, 100BASE-T1, and 1000BASE-T1. Some of the parameters presented depend on the specification. Others on the implementation. They represent best practice values.

4.3.1.3 100 Mbps over 100BASE-TX

In 2012, Marvell and Micrel presented an alternative solution for using 100 Mbps Ethernet in automotive [55] [56], which is sometimes called “QUIET-WIRE”; a registered trademark by Micrel [57], now owned by Microchip. Micrel had been an early development partner of BMW for the diagnostic Ethernet interface in 2008 (see also Section 3.1.3.4), had AEC-Q100 qualified some of their devices originally intended for industrial use, and thus some experience with automotive requirements.

The solution proposed uses IEEE 802.3 compatible hardware, based on 100BASE-TX. The transmission is consequently dual simplex, i.e., using one wire pair for transmission and one wire pair for reception. Apparently, the original 100BASE-TX signal is passed through a different, better adapted filter and transmitted via a better balanced link [56] [58]. With this, the output signal ceases to be strictly 100BASE-TX compliant, but in return shows promising EMC results.[11] The original implementations shown, used planar transformers, which together with two additional capacitors also functions as a filter [56] (Figure 4.31).

Reference [56] shows example test results achieved with the 150 Ohm method; results from a DPI test have also been presented. As is explained in more detail Section 4.1.3, the 150 Ohm method and the DPI test are a good starting point to assess the suitability of a semiconductor for automotive use. The proposed solution therefore certainly has automotive potential. As 100BASE-TX uses a significantly shorter scrambling than is used for 1000BASE-T and 100BASE-T1, the solution might show a different crosstalk behavior. Respective investigations have not come to the attention of the authors, though we expect it likely that sufficiently good results can be achieved.

Table 4.10 Comparison of PHY parameters between 100BASE-TX, 1000BASE-T, 100BASE-T1, and 1000BASE-T1, based on best design practices

Technology	100BASE-TX	1000BASE-T	100BASE-T1	1000BASE-T1
Channel length	100 m	100 m	15 m	15 m
PHY transmission	Dual simplex	Full duplex	Full duplex	Full duplex
X-level signaling	MLT-3 125MBaud	4D-PAM5 125MBaud	2D-PAM3 66.67MBaud	PAM3 750MBaud
No. of twisted pairs	2 (Cat 5)	4 (Cat 5e)	1	1
Required Nyquist bandwidth	62.5 MHz	62.5 MHz	33.33 MHz	375 MHz
Error correction	n/a	Trellis Coded Modulation	n/a	Reed-Solomon Coding
A/D conversion	5.5 bits ideal @ 125MBaud	7 bits ideal @ 125MBaud	7 bits ideal @ 66.67MBaud	Up to 8 bits ideal @ 750MBaud
DFE	16–24 taps	24 taps/channel	24 taps	Up to 128 taps
FFE	8 taps	12 taps/channel	8 taps	Up to 48 taps
NEXT cancellers	none	3 × 25 taps/channel	None	none
Echo canceller	none	160 taps	48 taps	150 taps
Critical path	• 3 input add • 3 input select • 1 slicer	• 4 input add-compare select • 3 input add • 5 input select • Branch metric compute	• 3 input add-compare select • 3 input add • 3 input select • 1 slicer	No information available at the time of writing
Normalized gate complexity	1	8	2	8
Additional features	Manchester coding provides spectral shaping	Partial response transmit filter	Transmit spectral shaping	Transmit spectral shaping

Note: See, e.g., [50].

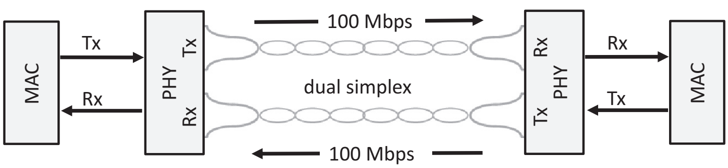

Figure 4.31 Key elements of the 100 Mbps over 100BASE-TX alternative.

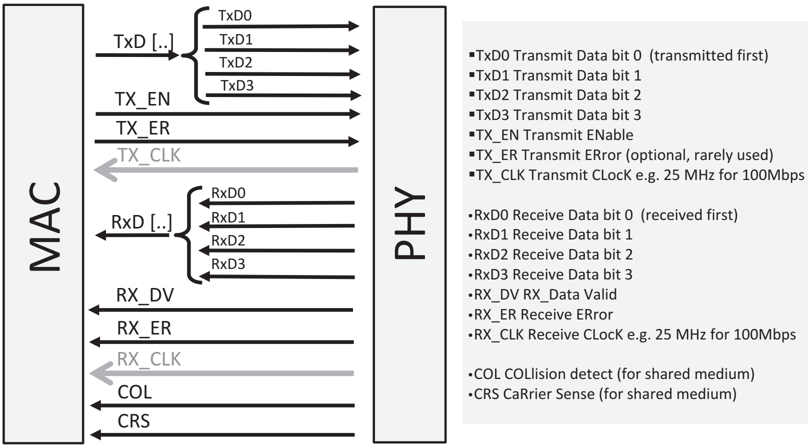

Figure 4.32 Definition of the MII interface.

An interesting use case for the proposed solutions is the diagnostic interface as standardized in ISO 13400 (see also Section 3.1.3.3), which requires the use of 100BASE-TX. If this interface can have a better EMC performance owing to the methods proposed with this solution, it might be possible to omit the disabling of the diagnostic interface during runtime of the car that is necessary today.

4.3.1.4 100 Mbps Ethernet over Media-Independent Interface (MII)

The xMII is an important element of Ethernet-based communication. It allows for the flexibility and scalability that makes Ethernet so attractive for automotive. Because of the xMII interface, different PHY technologies, even ones that use different media or speed grades, can be attached to the same switch and can be part of same Ethernet communication.

During the development of Ethernet for the diagnostic interface and the first investigations on the use of Ethernet in automotive with unshielded cabling, an unexpected property was discovered at the MII interface: In 10 Mbps and 100 Mbps Ethernet systems, always the PHY transceiver clock determines the clock for the communication with the MAC. It would have been more intuitive, if the system that sends the data determines the clock, i.e., the PHY transceiver when passing on received data to the MAC and the MAC in case it passes data on for transmission to the PHY. In case of a GMII, this is indeed organized this way (see also Section 4.3.2.1). When using an MII interface with 100 Mbps Ethernet, however, it is the PHY that determines the clock in both directions. The elements of the MII interface and the direction of the clocks are depicted in Figure 4.32. The MII transfers four bit words in parallel in each direction (see Figure 4.32), meaning that to achieve 100 Mbps the clock speed is 25 MHz. The MII was standardized with IEEE 802.3u and approved in 1995 [59].

Exactly this unexpected behavior of the MII clock can be used to enhance other transmission technologies with an Ethernet channel. As the PHY determines the clock, theoretically all clock rates are possible and it is not necessary to have synchronized clocks.

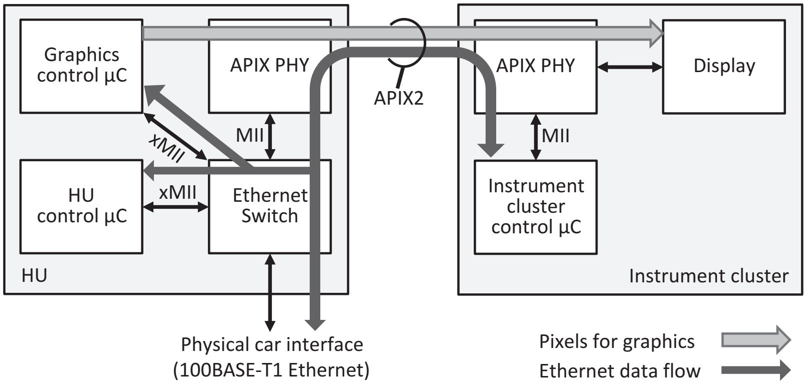

Figure 4.33 Example use case for an Ethernet over MII interface with APIX 2.

This means that the 100 Mbps MII is suitable to connect communication systems with completely different clock rates, provided the amount of transmitted data can be handled by the communication system.

One example of an automotive communication technology exploiting this is Automotive PIXel link (APIX, see also Section 2.2.6). With the generation of APIX 2, the technology received an MII interface and enabled an additional bidirectional Ethernet channel [60]. There are various technical solutions possible to achieve this; the one used for APIX is sideband modulation. The clock rate at the MII interface can be adjusted by the low-level settings of the APIX system and can be expanded to 25 MHz. The advantages are obvious. It is possible to have a unidirectional single hop, Point-to-Point (P2P) connection for, e.g., video data at a data rate of several Gbps, while at the same time the control data of bidirectional Ethernet can be seamlessly integrated into the vehicle's communication network via the MII interface.

Figure 4.33 shows an example of a HeadUnit (HU) that is connected to the instrument cluster. The task is to transmit graphics data at the same time and over the same physical connection as connecting the instrument cluster via the head unit to the in-vehicle Ethernet network/vehicle backbone network. The latter is achieved via the included Ethernet channel that connects to the network via an Ethernet switch (in this case in the HU), like any other Ethernet connection does. The microcontroller in the instrument cluster runs a standard TCP/IP software stack (see also Section 5.3). Data to be transmitted over the additional Ethernet link might be the engine speed, the velocity of the vehicle, lists of data from the HU, etc. For the communication partners in the Ethernet network the physical transmission technology is transparent. This is thus a good example for the "Ether" idea of Ethernet (see Section 1.2).

4.3.2 1 Gbps Ethernet

The bandwidth requirements in automotive keep increasing. Especially with the advent of automated driving, significantly more data need to be exchanged inside the car. Sensor data coming from various locations has to be made available at various different locations. For some of the data redundancy needs to be provided, as well in its generation as in its distribution within the in-vehicle network. Furthermore, with each next generation of mobile communication networks, the data pipes in and out of the car get bigger. Passengers deploy this for their infotainment, but also the car manufacturer can profit from this in respect to remote software updates and alike. The in-vehicle network has to support this, ideally with a flexible layout that should not be limited by bandwidth. Extending the Automotive Ethernet family with a 1 Gbps Ethernet PHY technology thus was inevitable (for data rates higher than 1 Gbps Ethernet, see Section 4.3.3.1). Section 4.3.2.1 discusses the PHY properties of the 1000BASE-T1 solution, while Section 4.3.2.2 gives a brief overview on a technology transmitting 1 Gbps Ethernet packets over Plastic Optical Fiber (POF).

4.3.2.1 1000BASE-T1

In March 2012, the IEEE 802.3 accepted a Call For Interest (CFI) on Reduced Twisted Pair Gigabit Ethernet [20]. The automotive market, with its increasing demands on networking and bandwidth, was identified as the driving market for the technology. Owing to the standardized xMII interfaces, Gbps was the next speed grade to adopt. Because of cost and weight requirements, twisted pair was the cable type to target for.

During the Study Group (SG) phase the automotive requirements were discussed in more detail. The whole range of topics, from temperature requirements, to EMC, to quiescence current, to diagnostic capabilities, to PHY latency, to crystal accuracy, to life time, to wake-up and channel requirements were presented in order to create a better understanding for the new industry in IEEE (see, e.g., [61], [62], [63]) and to help creating suitable objectives. In November 2013, IEEE 802.3 approved the objectives and the SG to become a Task Force (TF) [64].

The final technology [12] differs significantly from 100BASE-T1 (and 1000BASE-T); not only in the way the signals are handled exactly (especially in the PCS), but also because the specification additionally includes (optional) autonegotiation, (optional) Energy-Efficient Ethernet (EEE), and an Operation, Administration, Management (OAM) channel. Figure 4.34 gives an overview on the different building blocks. Instead of a separate block for PCS transmit enable like for 100BASE-T1, the 1000BASE-T1 PCS includes on the PCS side a separate block for the OAM. The 1000BASE-T1 PMA holds the PHY control, the link monitor, the PMA transmit and receive, the clock recovery, and an additional link synchronization block. For the signals exchanged between the PMA and PCS, 1000BASE-T1 includes extra information on the Low-Power Idle (LPI) status that is needed when the optional EEE has been implemented. Also, 1000BASE-T1 does not only use the loc/rem_rcvr_status values but also loc/rem_phy_ready.

In the following, this section first explains the different elements of the PCS and then of the PMA, adding some information on autonegotiation and EEE. It starts with the PCS transmitter elements as shown in Figure 4.35.

1 **80B81B encoding:** The 80B81B function encodes the information coming from the GMII, i.e., the TxD, TX_EN, and TX_ER into groups of 81 bits (referenced

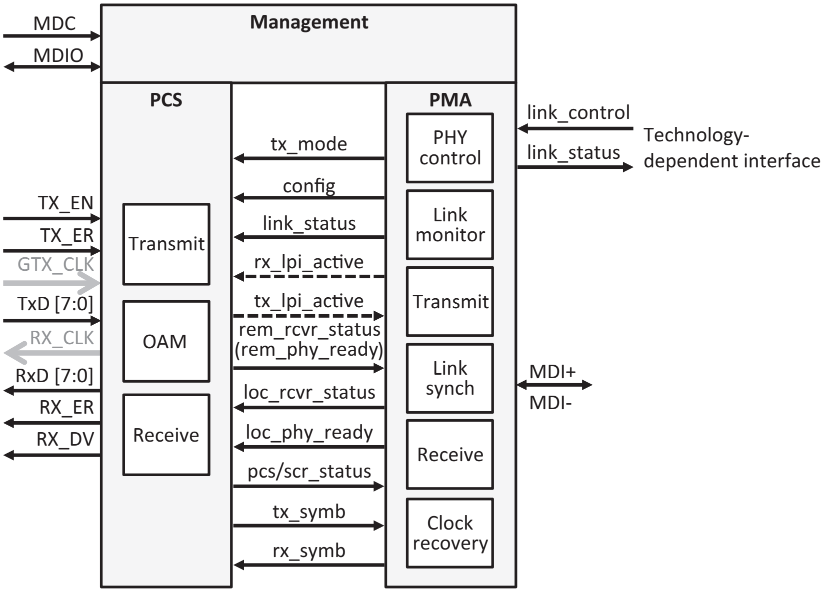

Figure 4.34 1000BASE-T1 building blocks (optional signaling for EEE in dashed lines

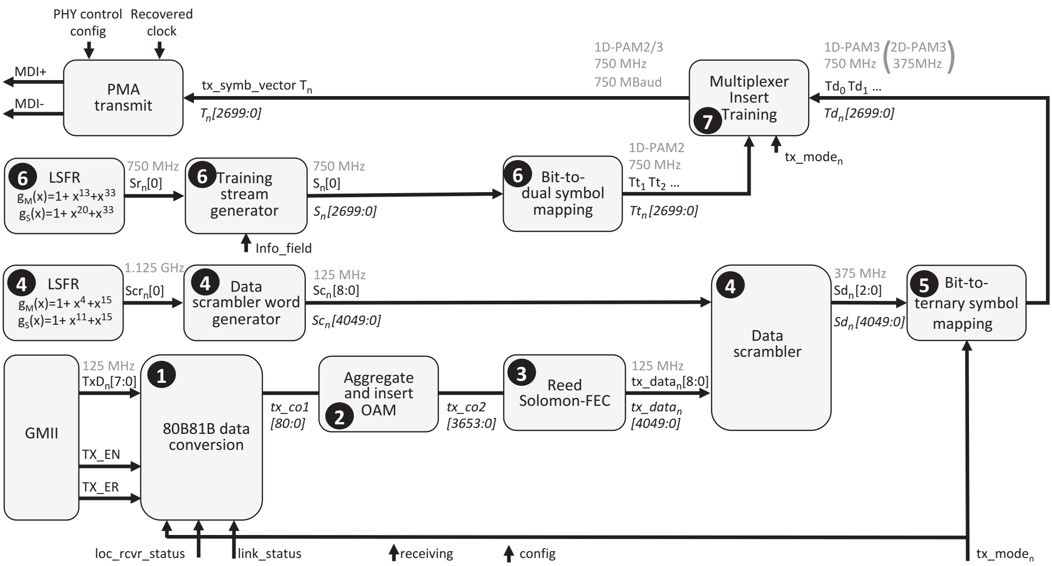

Figure 4.35 PCS transmitter elements in 1000BASE-T1 (without provision for link synchronization, autonegotiation or EEE). The output of the block is depicted with different data groupings [x:0]. The one above the arrows correlates to the frequency given, the one below in italic describes the block of data that belong together.

"tx_co1" in Figure 4.35). This encoding depends on the tx_mode and whether the group contains data only or also control information. For normal data transmission in tx_mode = SEND_N and TX_EN = 1 and TX_ER = 0, simply ten octets of TxD[7:0], i.e., 80 bits, are grouped together and prefixed by "0." In case of tx_mode = SEND_I, the 1000BASE-T1 specification is not specific. It is expected that the 80B81B function groups the same data it would use in case the system is idle during SEND_N (i.e., TX_EN = 0 and TxD[7:0] = 0).

In case the 81-bit group needs to contain control information (e.g., when TX_EN = 0 → normal idle, when TX_EN = 1 and TX_ER = 1 → error, when the loc_rcvr_status is not OK, or during Low-Power Idle (LPI) in case of EEE) the prefix bit is "1." The first four bits following indicate, where to locate the first 3-bit control code in the group "tx_co1," while the fifth bit indicates whether that code is the final control code in the block ("0") or whether more control information follows ("1"). Depending on the location indicated, the next bits are either the 3-bit control code, or data octets TxD[7:0] up to the control code, which can then be again followed by data octets.

2 **Aggregate and insert OAM:** 45 of the bit groups "tx_co1" are aggregated into one larger block of data (referenced "tx_co2" in Figure 4.35). Additionally, 9 bits of the 1000BASE-T1 OAM are added, making each block consist of $45 \times 81 + 9 = 3654$ bits. The OAM may be used for exchanging messages intended for the management, e.g., for monitoring the link health or supporting partial networking (see Section 6.3.3). Its deployment is optional, unless EEE is supported. Then the OAM has to be used – at a slower pace – to monitor the link health. If the OAM is not used in this PHY, nine "0" are added to each "tx_co2" block instead. If the link partner does not support the OAM, the nine bits transmitted are static.

An OAM frame consists overall of 12 bytes plus 12 parity bits, i.e., $12 \times (8 + 1 = 9)$ bits. One 9-bit group (one byte information, one parity bit) is included in every block of data "tx_co2" meaning that twelve "tx_co2" blocks are needed to transmit one OAM frame. The first two bytes of every OAM frame have set uses (see [12] for details). The following 8 bytes can be defined by the implementer, while the last two bytes contain a 16 bit CRC.

3 **Reed-Solomon FEC:** 1000BASE-T1 works with a lower SNR margin than 100BASE-T1. In order to maintain a target BER $<10^{-10}$, 1000BASE-T1 requires an FEC (that 100BASE-T1 does not have). The FEC defined uses a shortened Reed-Solomon code that operates on 9-bit symbols. The code used is a (450, 406) code meaning that it encodes $3654/9 = 406$ information symbols and appends $450–406 = 44$ parity symbols of 9-bit each at the end of each bock. It thus adds $44 \times 9 = 396$ bits and the output block length of tx_data increases to $3654 + 396 = 4050$ bits. The 44 parity symbols allow a correction of up to 22 symbol errors.

4 **Scrambling:** In order to improve the EMC performance, the thus created block of tx_data[4049:0] is scrambled, i.e., bitwise exored with a pseudo-random bit sequence. This sequence is generated with an LFSR) using the master and slave polynomials as defined in Equation 4.6. This sequence does not receive any further processing but directly feeds into the scrambler. The extra data scrambler word

generator block in Figure 4.35 has only been introduced in order to point out the different bit grouping at different frequencies. While the 9-bit group $Scr_n[8:0]$ has to have the same frequency as the 9-bit output from the FEC, the original LFSR has to run at nine times the speed, i.e., at 9×125 MHz $= 1.125$ GHz.

The initial starting sequence in the scrambler is up to the implementer (other than all "0"). These initial values are transmitted to the receiving unit with the info_field exchange during training (see point 6). This is also different from 100BASE-T1/1000BASE-T, where the initial values of the scrambler are derived from the received data sequence that is loaded into the scrambler during link acquisition.

$$g_M(x) = 1 \oplus x^4 \oplus x^{15}$$
$$g_S(x) = 1 \oplus x^{11} \oplus x^{15} \quad (4.6)$$

5 **Bit to ternary symbol mapping:** For processing, the FEC had grouped the data into blocks of nine bits. For the bit to ternary symbol mapping, these are further divided into blocks of three $Sd_n[2:0]$. Like for 100BASE-T1 each triplet of bits is mapped onto two PAM3 symbols. Other than for 100BASE-T1, 1000BASE-T1 uses a Gray code for this (see Table 4.5 in Section 4.3.1.2). This Gray code mapping is used in case the tx_mode is in SEND_N or SEND_I. The standard labels the two output PAM3 symbols T[0] and T[1], which defines their order and eliminates the necessity to detect their correct order in the receiver, as is needed for 100BASE-T1.

Should the tx_mode be in SEND_Z, zero voltage will be put on the line, which corresponds to $\{T[0], T[1]\} = \{0,0\}$. Should the connection be in training, for which an extra tx_mode = SEND_T is introduced, an entirely different path and set of data is used, as explained with the next point. The 4050 bits that are part of one block of data after FEC and scrambling make 2700 PAM3 symbols after the 3B2T conversion.

6 **Training mode:** The training mode of 1000BASE-T1 has the purpose to support the receiver of the link partner in adjusting and in aligning to the block of transmit data. Its processing is entirely different from the data and idle stream processing. The blocks for the symbols sent during training have the same length, i.e., they also consist of 2700 symbols each. However, the symbols during training are PAM2 modulated, meaning that the bits are mapped directly onto the voltage levels. If the input bit $S_n[0] = 0$ then $Tt_n[0] = +1$. If the input bit $S_n[0] = 1$ then $Tt_n[0] = -1$. The overall bit sequence that comprises one block of training information thus consists of not only of 2700 symbols but also of 2700 bits.

To generate this training data a pseudo-random bit sequence $Sr_n[0]$ is generated with help of an LFSR), which is then used to scramble the information that goes into the training stream $S_n[0]$. The polynomials for generating $Sr_n[0]$ are defined in Equation 4.7 (they are the same as the polynomials used for 100BASE-T1 and 1000BASE-T).

$$g_M(x) = 1 \oplus x^{13} \oplus x^{33}$$
$$g_S(x) = 1 \oplus x^{20} \oplus x^{33} \quad (4.7)$$

For the sequence generated during training $S_n[0]/S_n[2699:0]$, each 2700-bit block of training data is split into 15 partial frames of 2700/15 = 180 bits each. The first bit of all partial frames except the last, is inverted. The first 96 bits of the 15th partial frame is exored with the content of the info_field. Equation 4.8 describes the generation of the training stream $S_n[0]/S_n[2699:0]$.

$$S_n = \begin{cases} Scr_n \oplus \text{info\_field}_{(n \bmod 180)} & \text{if } 2520 \leq n \bmod 2700 \leq 2615 \\ Scr_n \oplus 1 & \text{else if } (n \bmod 180) = 0 \\ Scr_n & \text{else} \end{cases} \quad (4.8)$$

The info_field consists of $12 \times 8 = 96$ bits. The first three bytes always have the same predefined values that serve as a start frame delimiter. The next three bytes contain the Partial PHY Frame Count which indicates the running number of the partial frame. The slave synchronizes its partial frame count to the master's. The following one byte message includes information on the PMA state (two bits that are 00 for training or 01 for countdown, see Figure 4.30, and used in the training symbol path only), the loc_rcvr_status, and whether the master is ready for the slave to transmit ("en_slave_tx") or whether the slave completed the timing lock. The next three octets depend on the state during training. In case of PMA = 00 (training state) the octets include information on OAM and EEE capability and on the scrambler starting sequence for the data path. In case of PMA = 01 (countdown state), the three bytes include the partial frame count that indicates when the system changes from PAM2 to PAM3 modulation. In both cases, the last two bytes of the info_field contain a CRC. The correct reception of the info_field is decisive for the start-up in 1000BASE-T1. It is therefore transmitted at least 256 times, to allow detection by the link partner.

7 **Multiplexing:** Depending on the tx_mode the multiplexer will put the training (tx_mode = SEND_T) or the idle/data streams onto the channel (tx_mode_ SEND_I/SEND_N).

Unlike 100BASE-T1, the 1000BASE-T1 specification foresees the polarity change in the receive path only, independent from whether this is the master or slave unit. The PCS transmitter therefore does not need to include a polarity change unit; only the PCS receiver does.

There is one additional aspect, the PCS transmitter has to take care of. It comes from the fact that the MII of 100BASE-T1 handles the clock differently than the GMII of 1000BASE-T1 does. In case of an MII, both the receive and transmit clocks are determined in the PHY and passed up through the MII (see also Figure 4.32 in Section 4.3.1.4). Thus, there is no risk of misalignment in the PHY. In case of a GMII this is different. Only the receive clock is determined by the PHY and passed up through the GMII. On the transmit side, the GMII passes a clock GTX_CLK down to the PHY. The PCS transmitter has to provide for a potential misalignment between GTX_CLK and the PHY internal (receive) clock. How this is done, is left up to the implementer.

In principle, the PCS receiver just performs the tasks of the PCS transmitter in reverse order. Once the training has been completed and the data blocks are correctly aligned, the receiver demodulates the PAM3 symbols into bits, descrambles the data, decodes it with potential error correction, extracts the OAM, potential control data (esp. RX_DV and RV_ER), and the data octets RxD[7:0], which are then passed to the GMII. The 81B80B conversion removes the appended bit and marks a block as erroneous (RX_ER = 1), if a block marked to hold control frames, either points to an invalid location or contains a control not defined, or if the FEC flagged a block to hold errors that cannot be corrected.

In order to be able to receive properly, the PCS receiver, however, first has to perform its share of the synchronization process. It has to start with potentially correcting the polarity and locking its training descrambler. How this can be done is not defined, but it is up to the implementer how the receiver makes use of the incoming PAM2 training stream for this. Similar to what is being done in 100BASE-T1, the receive scrambler can load the received data into the LFSR registers until the output and input signals of the scrambler are the same. However, the 1000BASE-T1 implementation also has to take the processing of the training data as described in Equation 4.8 into consideration.

Once the descrambler has been locked, the PCS sets the scr_status to OK. With the scrambler locked, the PCS will also know when one 2700-bit data block starts, i.e., it will have achieved frame synchronization. It can then extract the information from the info_field essential to complete the process. Especially it needs the starting input for the scrambler in the data path and the partial frame count for when the PHY control goes into countdown (see PHY control description further below).

On the transmit side, the **PMA transmitter** has the task to put the PAM2/3 symbols onto the channel such that the electrical specifications like the PSD mask limits can be met. This can comprise filtering, D/A, and subsequent analog filtering, as described for 100BASE-T1. Because also 1000BASE-T1 uses one UTP cable only, a hybrid is needed to allow reception and transmission simultaneously on the same line and to be able to extract the PAM2/3 symbols from the incoming signals. On the **PMA receive** side also for 1000BASE-T1 a loop-timing concept is foreseen for **clock recovery** (see Section 4.3.1.1 or 4.3.1.2 for details). Like the 100BASE-T1 PMA, the 1000BASE-T1 PMA receive function will include equalization and cancellation of the echoes its own transmit signal imposes on the receive signal. Quite a few differences exist in the link-acquisition process the PHY control describes (see also Figure 4.36). It is described in more detail below. The **link monitor** of 1000BASE-T1 is not much different from the one in 100BASE-T1, except for that the hysteresis used in 100BASE-T1 is replaced by the integration of the minwait and maxwait timers into the process.

Figure 4.36 depicts the flow charts for master and slave **PHY control** including some correlation in timing they might have. The process is started with enabling the link_control under the condition that the link synchronization (see more below) or, if available, the autonegotiation has been completed. Both master and slave start with the slave silent SEND_Z mode. The master immediately changes into training mode SEND_T and starts sending the PAM2 training data. With starting to transmit, the master adjusts to receive data, e.g., by adapting its echo canceller. Once the master is sufficiently prepared (which is up to the implementer to decide), it sends en_slave_tx = 1 to

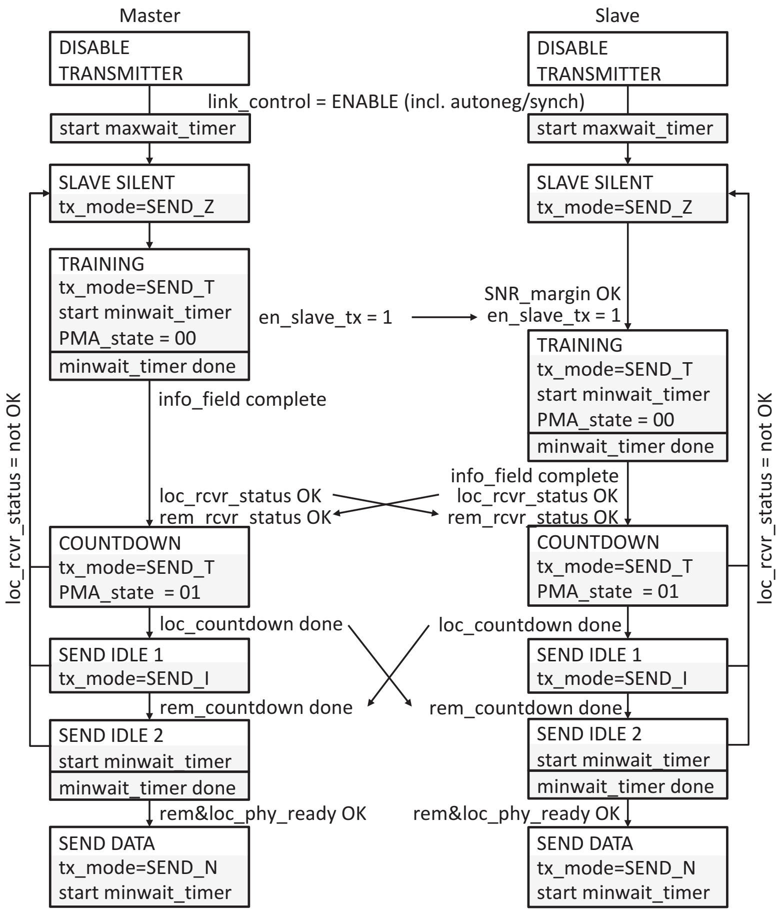

Figure 4.36 PHY control process for 1000BASE-T1.

the slave within the info_field encapsulated in the training stream. The slave, which will have made use of the master's training stream to adapt its AGC, FFE, etc. to determine that it has sufficient SNR_margin, can then also go into SEND_T and starts sending its own PAM2 training data.

Once both loc_rcvr_status are OK and each link partner has conveyed this to the other, the units can go into countdown. The countdown step is necessary to ensure correct alignment to the 2700-symbol blocks and exact time of the PAM2 to PAM3 switchover. As soon as the local countdown has been completed the system switches into SEND_I where it starts the minwait_timer as soon as it perceives the remote countdown to have been completed, too. From Figure 4.30 it could be concluded that this happens the same time. However, there is no time synchronization between master and slave. The countdown values might actually be different as might be the absolute starting time. An exact alignment of the changeover times in master and slave is therefore unlikely. Once loc/rem_phy_ready are OK, the transmission switches from sending idle

SEND_I to sending data SEND_N. Note, that the loc/rem_phy_ready is a new variable that 100BASE-T1 does not have. It was introduced because of the different PCS paths that are used during SEND_T and SEND_I/N. The loc_rcvr_status will have been set OK during SEND_T (or the PHY control will not change into SEND_I), relying on the training path of the PCS only. The loc/rem_phy_ready status additionally ensure that also the data path of the PCS is ready.

The 1000BASE-T1 specification describes the optional use of autonegotiation. When autonegotiation is supported, the first step after power on is to determine the capabilities of the link partner and to agree on the set of capabilities to use. This does not only comprise the transmission speed/technology on the link to use – 100BASE-T1 or 1000BASE-T1 – but includes establishing which link partner is master and which is slave. Without autonegotiation master and slave status are determined by the management function or hardware configuration. The autonegotiation function will not be described in more detail here, as its use in the predetermined automotive in-vehicle networks is limited. However, next to negotiating and aligning on capabilities, the autonegotiation fulfills another important task: It provides for the first handshake between link partners and ensures that the states (tx_modes) are synchronized before the units start with the link-acquisition process as defined in the PHY control sequence (see Figure 4.30).

In case autonegotiation is not supported, these tasks have to be performed in a different way. 1000BASE-T1 therefore specifies a separate **link synchronization** process. The link synchronization is used by the link partners to discover the other link partner and to synchronize the units into the same state SEND_S/SEND_Z; not only once at the beginning, but every time the link acquisition needs to restart. This process provides for a proper handshake and prevents that the units end in a dead lock situation, where, e.g., because one unit lost the link and the other notices this with a time delay, the link partners try to reacquire the connection in different stages [65]. In case of autonegotiation, the autonegotiation process would be restarted and thereby ensure a defined starting state. In SEND_S, the link synchronization deploys a special pseudo-random sequence derived with an LFSR using the master and slave polynomials as defined in Equation 4.9. These bits are PAM2 modulated before being forwarded to the PMA. This sequence is transmitted independently of any 2700 symbol blocks of the transmit states SEND_T, SEND_I and SEND_N, but repeats after every 255 bits = symbols. The transmission in SEND_S is half-duplex, i.e., for the handshake, first the master transmits SEND_S, then the slave.

$$P_M(x) = x^8 \oplus x^4 \oplus x^3 \oplus x^2 \oplus 1$$
$$P_S(x) = x^8 \oplus x^6 \oplus x^5 \oplus x^4 \oplus 1 \quad (4.9)$$

1000BASE-T1 provides for the (optional) use of EEE (see also Section 4.4.5). The idea is to significantly reduce the power consumption of a PHY when it has a link established but no data to send. The Low-Power Idle (LPI) state can thus only be entered during normal data mode SEND_N (and if both units support it). Next to the normal data transmit state, EEE knows QUIET and REFRESH. Per specification, the LPI is initiated upon TX_EN = 0, TX_ER = 1, and the transmit data group TxD[7:0] = [00000001] = 0x01 at the GMII interface. The PHY will then first send a sleep frame, i.e., a whole block of 2700 symbols of low-power idle symbols (TX_EN = 0, TX_ER = 1, TxD = 0x01). During QUIET no voltage is put on the channel and the transmitter as well as the receiver can power down parts of their circuitry. In order not to lose synchronization, the PHY periodically sends refresh blocks, which are generated from the scrambler output in the data path that is being PAM3 encoded. To wake the system up again, the PHY will send full block of idles (i.e., zero data FEC encoded and scrambled) at the next possible wake window (see below) and thus end the LPI mode. The EEE process is similar to what has been defined for 10GBASE-T other than that 1000BASE-T1 uses one UTP cable only, that refresh can use the data path and that the data blocks in 1000BASE-T1 are longer than for 10GBASE-T [66].

1000BASE-T1 EEE also functions on the basis of 2700-symbol blocks. A period of quiet and refresh lasts 16 blocks = $16*2700*3/2$ bits = 180 bits × 360 partial frames, with 354 partial frames QUIET and six partial frames REFRESH. This reduces the active transmit time to 6/360 = 1.66%. The refresh from the slave is sent at an offset of 360/2 + 15 = 195 partial frames from the master refresh. When sending wake blocks, this is possible every second block boundary for the master and every alternating second block for the slave. The wake signal is an entire 2700-symbol block of idles. The wake-up can be initiated by the GMII when data to send is available, or when the PHY perceives that it cannot maintain a good enough SNR during LPI. Once the system has ended the LPI, it can attempt to recover the link during normal power-up mode.

4.3.2.2 1000BASE-RH

The strict layering and availability of xMII interfaces is the basis for the scalability and attraction of (Automotive) Ethernet. Provided both ends of a link use the same technology and have the respective xMII interface, the technology as such is transparent to the network. Thus, many Ethernet standards have been developed for different media, like coaxial, twisted pair, twin ax, backplane and fiber (see also Section 1.1). One of the first 802.3 standards (IEEE 802.3d) was for a fiber-optic interrepeater link and standardized in 1987. 10BASE-F was standardized in 1993 [67].

Optical transmission, at least over Glass Optical Fiber (GOF), has the advantage of extremely low attenuation and delay and therefore supports either higher data rates and/or longer distances [68]. However, for automotive use, GOF is not (yet) suited. It is seen as too expensive and difficult to handle, while the long reach is not needed. The automotive industry has always preferred Plastic/Polymeric Optical Fiber (POF), which is mechanically less sensitive than GOF. For the IT/communications industry POF is not of interest, because the reach significantly shorter than for GOF [69]. For automotive use the short reach is sufficient, so the industry can profit from the robustness (unreceptive to EMC in the harness, see also Section 2.2.4) at a price level between UTP and STP [70].

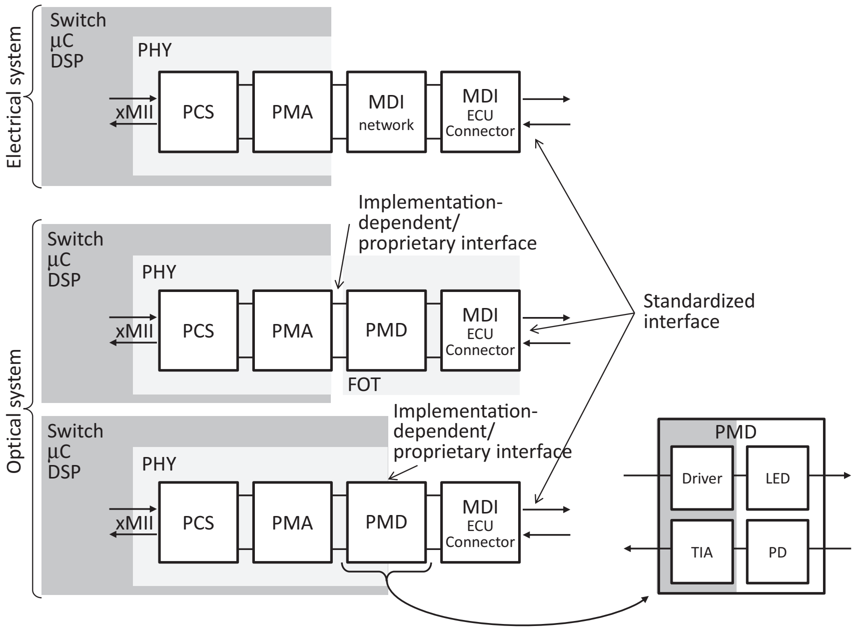

Figure 4.37 Transmitter elements and their possible integration in semiconductors in an optical system in comparison to an electrical system. TIA = TransImpedance Amplifier.

Costs and reliability are the main criteria when deciding between UTP and POF Ethernet. A few other considerations are listed below [70]:

1 The medium POF in automotive use traditionally supports temperatures from −40°C up to +95°C only. Some car manufacturers require up to +125°C. Higher temperature POF can be provided, however at a noticeably higher price level.
2 For optical systems it is not possible to transmit the power over the data line, like it is the case for electrical systems (see also Section 4.4.3). For optical systems power always requires separate wiring.
3 Electrical PHYs have fewer restrictions when being integrated into switches or microcontrollers than optical PHYs. The reason for this is depicted in Figure 4.37. The CMC and DC isolation capacitors that make the MDI network between the output of an electrical PHY chip and standardized output of the MDI/connector are more or less transparent from an electrical point of view. This means that PHYs are developed toward the standardized output. Optical transmission, in contrast, requires a media conversion in the Physical Medium Dependent (PMD), whose Light Emitting Diode (LED) and Photo Diode (PD) cannot be integrated in a CMOS semiconductor. Most commonly the PMD and the MDI get integrated in a Fiber Optical Transmitter (FOT). This means that the PHY is developed to a proprietary output and PHY and FOT vendors have to align their products during development, which limits the number of FOTs usable in each case. Thus, anyone integrating an optical

PHY into a switch or microcontroller also restricts their parts to specific FOTs or related products.

4 Optical systems are more demanding in their mechanical handling than UTP systems. The bending radius is limited. The tensile force is strictly limited. The optical harness element needs to be preassembled, while it is essential it receives complete protection against contamination, cuts, or corrugations on the inner sheathing when being installed in the cars.
5 In case of problems in the field, at least BMW engineers find the maintenance of optical solutions to be more difficult than for UTP but better than for STP cables.

So, in March 2014 a CFI was passed for 1 Gbps Ethernet transmission over POF [71], which mentions three application areas: Consumer networking, automotive networking, and professional networking. Under the assumption that Gigabit Ethernet for Plastic Optical Fiber (GEPOF) is more expensive than the electrical 1000BASE-T1 in automotive, it is a potential alternative for special use cases where UT(S)P is not possible, either owing to link distance or to EMC criticality. In consumer/home networks POF is seen as an alternative when neither wireless technologies provide sufficient quality (limited reach owing to building fabric or size), nor Power Line Communication or UTP cables do (for EMC reasons). In professional networks POF is of interest in EMC sensitive areas or when a galvanic isolation is needed.

Optical transmission systems started off with using red light LEDs – red LEDs were most commonly available and lower priced than other colors[12] – and the use of a straight forward Non Return to Zero (NRZ) modulation. However, with this concept it is not possible to achieve a data rate of 1 Gbps over POF for two reasons: 1. The LED cannot emit light fast enough [72]. 2. The attenuation of POF is too large. For POF the attenuation varies between 10 and 1000 dB/km, depending on the material used [73]. Absorption and scattering are the reason for the loss, and they are caused by impurities in the fiber. With the used SI-POF the maximum bandwidth with red LEDs and NRZ modulation is only about 75 MHz (End-to-End@-3dB) or 150 MHz with improved driver preemphasis [74].

1000BASE-RH therefore uses an intelligent multilevel coding; which includes a PAM16 (for the header) and PAM8 (for the payload) modulation at 162.5MBaud and a Tomlinson Harashima precoding [75] [76] in order to mitigate error propagation in the DFE [77]. As a consequence of the multilevel modulation, analog reception of the optical information is necessary. Like 100BASE-TX, this solution uses dual simplex transmission (in a "full-duplex" Ethernet network), i.e., one link for transmission and one for reception of data. The release of the thus developed IEEE802.3bv/1000BASE-RH standard is expected for early 2017.[13]

4.3.3 Other Data Rates

One of the main advantages of Automotive Ethernet is its scalability. An in-vehicle Automotive Ethernet network can comprise links with different data rates depending on the local requirements, while this is being completely transparent to the higher layer

Table 4.11 Required data rate in Gbps for uncompressed data depending on the horizontal and vertical resolution (Hres, Vres), the bit depth, and the frame rate (fps)

			Bit depth (Gbps)				
Hres	Vres	fps	8	12	16	20	24
1280	720	30	0,22	0,33	0,44	0,55	0,66
1280	1080	30	0,33	0,50	0,66	0,83	1,00
1920	1080	30	0,50	0,75	1,00	1,24	1,49
3840	2160	30	1,99	2,99	3,98	4,98	5,97
1280	720	60	0,44	0,66	0,88	1,11	1,33
1280	1080	60	0,66	1,00	1,33	1,66	1,99
1920	1080	60	1,00	1,49	1,99	2,49	2,99
3840	2160	60	3,98	5,97	7,96	9,95	11,94

implementation. After the completion of the 100BASE-T1 and 1000BASE-T1 standards the question therefore was, what other data rates are needed for Automotive Ethernet? The following sections will describe the situation for data rates higher than 1 Gbps (see Section 4.3.3.1) and for data rates lower than 100 Mbps (see Section 4.3.3.2).

4.3.3.1 Ethernet for Data Rates Higher Than 1 Gbps

The amount of data transmitted inside cars has increased drastically in the last years and is continuing to increase. One of the main drivers for data rate is video and graphic data, i.e., camera and display usage. Table 4.11 shows data rates needed for various different resolutions and frame rates in case of uncompressed video and graphic data. As can be seen, 1 Gbps data rate is easily reached, especially in case of a frame rate of 60 fps.

Naturally, most in-vehicle Electric and Electronics (EE) architectures will try to avoid transmitting multi-Gbps video streams throughout the network. If possible, data will be compressed or powerful Point-to-Point (P2P) pixel links can be used, were the communication is locally restricted. However, neither compression nor P2P is always possible; not in case of specific image processing, specific video architectures, nor if redundancy requires duplication of sensor data including video.

Another source of high data rates is the communication between the car and the outside world, i.e., mobile data communication and in parts digitalized media broadcast. The development of the mobile data communication over time is shown in Figure 4.38. It can be seen that the achievable data rates have continuously increased over the decades and that data rates larger than 1 Gbps can be expected in the near future. Car manufacturers will want to use that data rate as well for more services to their customers. The data will need to be distributed from the car antenna into the vehicle.

As for digital radio/TV/media broadcast: Broadcast is being digitalized worldwide in order to save energy and use the bandwidth more efficiently [80] [81]. To achieve the latter, digital broadcast standards use compression methods. However, broadband reception methods and/or decompression potentially increase the data rate to be transmitted significantly. Last but not least, a trend in the EE architecture is to concentrate

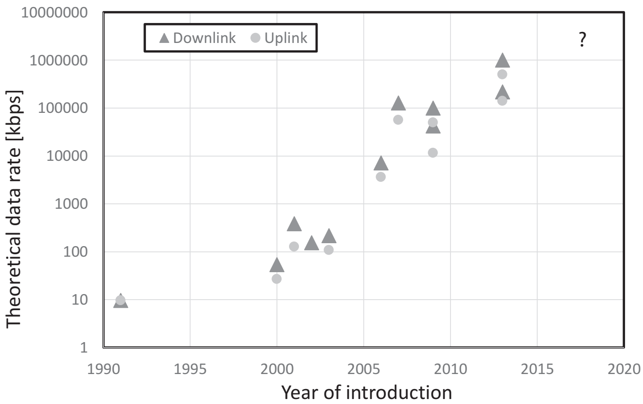

Figure 4.38 Development of (theoretical) data rates in mobile communication over time [78] [79].

the computing in few but powerful computing platforms inside the car (see also Chapter 7 or [82]). Communication between those can also make use of higher data rates.

So, rather sooner than later, the in-vehicle network requires an Automotive Ethernet PHY that supports more than 1 Gbps data rate and a respective CFI was passed in IEEE 802.3 in November 2016 [83]. One of the most challenging aspect of the project is represented by the definition of the channel. Other than with 1000BASE-T1, not only the number of pairs but data rate and media – UTP, coax, STP, optical – have to be selected. Results of respective investigations prior to the IEEE efforts have been published in, e.g., [84] [85] [86] [87].

4.3.3.2 Ethernet for Data Rates Lower Than 100 Mbps

It seems to be a natural instinct, when looking at the next step, to look at "more." However, in the discussions around the introduction of Automotive Ethernet with different car manufacturers it became evident, that for many use cases, a data rate of 100 Mbps actually provides too much data rate. Naturally, too much means the solution is not cost and energy efficient enough, especially in light of other in-vehicle networking technologies that support lower data rates like FlexRay or CAN FD. Those can at least cover (shared) data rates of a few Mbps (see also Sections 2.2.2 and 2.2.5).[14] Thus, in case, e.g., only 5 Mbps are needed it is expected that an Ethernet technology is cost competitive to existing non-Ethernet solutions, while it can still seamlessly integrate into the Ethernet network and reduce the number of gateways [88].

Therefore, the main concern for the automotive use case was, whether a system cost reduction of 50%, when compared with 100BASE-T1, could actually be achieved (the technical feasibility was never questioned). First estimations deemed this possible [89] [90], and efforts were therefore started to also standardize a lower cost 10 Mbps automotive suitable Ethernet PHY at IEEE 802.3. This effort happily coincided with the industrial automation community also looking for a 10 Mbps Ethernet solution. While their main concern was reach (1000 m), it was still fitting to have both efforts join.

A CFI to start a respective study group thus passed in July 2016 [91]. The request to go to task force was successfully placed in November 2016 [92].

The objectives for the thus numbered IEEE 802.3cg (expected to be called 10BASE-T1) project comprise the support of 10 Mbps Ethernet data over single balanced twisted pair cabling with full support of the automotive and industrial environments (e.g., fast start-up, intrinsic safety). The objectives foresee two link segments: one of 15 m with four inline connectors, one of 1000 m with ten inline connectors. Optional features for the project were single-pair autonegotiation, EEE, and power distribution over the data line [93].

It was intentionally left open, whether the project would support a bus topology (shared media). Especially for the automotive use case with the 15 m link this promises additional cost advantages important for the adoption. At the time of writing it was not foreseeable whether this would indeed be included in the project or not. Another important aspect was timing. The sooner the project can be completed the better. With the long introduction cycles in automotive, earlier availability has the potential to give yet another boost to the success of Automotive Ethernet.

4.4 Automotive Ethernet and Power Supply

Power supply and energy consumption are important topics when designing a car. A reliable power supply is a prerequisite for the functioning of modern cars. At the same time, consumers are not only increasingly aware of the fuel consumption (costs), but also legislation worldwide is setting more stringent targets for CO_2 reduction.[15] In 2013, transportation was responsible for 23% of the worldwide CO_2 emissions and thus the second largest CO_2 producing sector after electricity and heat generation (which had a 42% share [94]). Within transportation, road vehicles account for 75% of the emissions, i.e., for about 17% of the overall emissions. Over the last decades, the CO_2 emissions in all sectors have significantly grown; for road vehicles by 66% between 1990 and 2013. This actually shows that vehicle manufacturers have already reduced the emissions per vehicle, as the number of registered vehicles about doubled in the same timeframe [95] [96]. However, more efforts are required and thus all elements inside the car, also the in-vehicle network, are scrutinized in respect to their power balance.

There are many different factors that determine the energy consumption and CO_2 emission of a car. In the environment of in-vehicle networking the factors of interest are weight as well as power consumption of the network, when in use and when not in use. As has been discussed, the wiring harness is the third most heavy (and the third most expensive) component inside a car [20]. The possibility to use 100BASE-T1 with UTSP cabling was an important motivation to introduce Automotive Ethernet, as it has cost and weight advantages over other cabling options (see, e.g., [31]). However, while it is of interest to keep the weight of the communication network low, BMW internal investigations have shown that in a typical car today the supply network has the largest share in the weight of the harness.[16]

To explain the relation between power supply and Automotive Ethernet, Section 4.4.1 first of all shows what a power supply network consists of with all its relevant elements, before discussing the impact of the power supply on the in-vehicle communication in Section 4.4.2. Once these basics have been established it is possible to discuss Automotive Ethernet related approaches that can affect the power consumption. Sections 4.4.3 and 4.4.4 discuss methodologies to save cabling by either transmitting power with the Ethernet communication or by putting the communication over the power supply cables. In this context Power over DataLine (PoDL) has been standardized for Automotive Ethernet in IEEE 802.3bu.

Concerning the power consumption of the network itself, various aspects are relevant. For one, it is the complexity of the communication technology itself. There is a trade-off between the cabling and the PHY; not only in costs as discussed in Section 4.2.4 but also in respect to power consumption. A less robust cable will likely require more complex and therefore more power consuming signal processing in the PHY. However, this is not subject of this section, as the technologies have been defined. The power consumption needed in case of an active link is now up to the implementation and a differentiating feature for PHY solutions on the market. This section focusses on the possibilities to reduce power when the link or even ECU behind the link are not active. Section 4.4.5 thus discusses the use of the IEEE 802.3az standard on Energy-Efficient Ethernet (EEE) for automotive that provides a solution to save power on an Ethernet link in case of link inactivity. Section 4.4.6 describes possibilities to not only save power in the PHY, but in the complete ECU, with sleep and wake-up mechanisms, as they have been specified in the OPEN Alliance and incorporated in ISO 21111.

4.4.1 Elements of the Power Supply Network

Figure 4.39 gives an overview on the elements relevant for the power supply network. The different elements are explained in detail below.

- Traditionally, cars are built with a conductive **metal body (1)**. This is very important, because the continuous metal body of the car is a reference point to which the negative pole of the car's battery is connected to. The ground (= negative pole of the battery) is thus distributed all over the car's body and the ECU's ground can be connected to it on the most direct path available. In contrast, the positive pole requires distribution in a more complex fashion and requires to be well isolated from the car's body.

  With more digitization of vehicle functions and thus more electronics in the vehicles, the power consumption increases and, with it, the currents that need to be distributed. This causes new challenges, as poor design can lead to EMC disturbances in form of very strong magnetic fields due to the high currents needed. To twist also power supply cables where both the ground and plus cables are available helps to reduce impairments in respect to EMC. Additionally, the body structure is changing gradually. This is mainly due to new materials being used with mixed in plastic or bonded carbon fibers. This impacts the current flow and can have a negative impact on the overall EMC behavior of the vehicle.

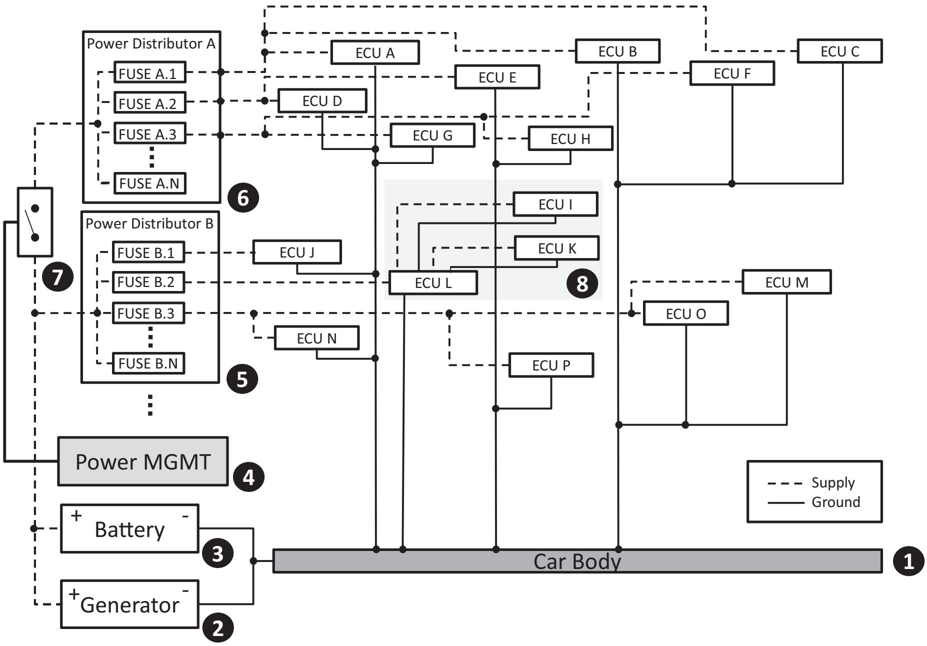

Figure 4.39 Example of a typical power distribution in a car. In this figure, "ECU" is used synonymously for all electrically powered units in the car, whether they are ECUs, sensors, or actuators.

- **Electric generator (2)** has the purpose of recharging the car's battery. The charging system in modern cars is extremely optimized. Especially in hybrid or electric vehicles the power supply for the standard device uses a high-voltage system of approximately 400 V via DC/DC converters. A charge controller ensures that the electric generator charges the **battery (3)** correctly and makes the system stable. Thus power supply during vehicle operation is secured.
- The **power management system (4)** enables that the energy is distributed and supplied to the ECUs. It has to take care that ECUs, which do not require power in certain use scenarios are switched off from the power supply. This is also referred to as clamp management (see also Section 6.3.3). Some ECUs receive power only when the engine has been switched on (referred to as Clamp 15), other are always connected to the battery (Clamp 30, output of distribution box 5 in Figure 4.39), yet others are also connected to the battery but switched (Clamp 30g, output of distribution box 6 in Figure 4.39). Depending on the complexity of the vehicle, a number of **on/off switches (7)** are used for this (Figure 4.39 shows only one), which are traditionally classic relays. Today, also specialized semiconductors can be used for this purpose. In either case, if an ECU is switched off from the power supply by such switches, it cannot be switched on again based on anything that happens on the wire of the IVN technology. Since the ECU is no longer supplied with power, no electronics is active that can recognize wake-up patterns or other activity on the communication link.

Thus there is a strong interdependency between the sleep/wake-up of the power supply and the IVN-technologies. The simultaneous use of both has caused destabilized vehicle functioning in the past. This has two consequences. (a) The deployment of wake-up solutions provided today with traditional IVN-technologies is limited. (b) The power supply/wake-up systems are designed to be very hierarchical and need to be organized centrally. Note, that also battery drainage can be a result if shut down sequences of ECUs are not planned carefully. At the same time, with strict targets on CO_2 emissions, to use wake-up and shutdown of ECUs in cars is more important than ever.

- The **power distributors (5, 6)** ensure the distribution of the positive supply to the ECUs. As a rule, fuses are used on the distribution lines to prevent electrical fires and to protect the wires from overload. Power distributors are often spatially organized. There might, e.g., be a distributor for the front of the car, another for the rear or also one for the engine compartment. Semiconductors to replace fuses are unfortunately still too expensive today, but they would have many advantages like active driving of the power supply lines and additional diagnoses of fault currents by active measurements. Overall, the power supply will be more individualized and adjusted to the actual need in the future.
- Figure 4.39 additionally shows the option of **one ECU** L **feeding the power of other ECUs** I and K **(8)**. One option for realizing this is PoDL (see also Section 4.4.3). As said, PoDL can save power supply lines. Additionally, the power is provided over the almost perfectly symmetrical wire pair of the IVN technology, which has a positive impact on the vehicle EMC.

4.4.2 The Interconnection between Power Supply and Communication Technologies

Figure 4.40 illustrates the same scenario as in Figure 4.39 but with the focus on the communication. In this example the ECUs H, L, and M contain Ethernet switch ICs. The rest of the communication nodes are connected via PHYs only. Without the knowledge of the power supply network in the background it is very difficult to optimize the communication system in terms of start-up, shutdown, or wake-up. Simply speaking it is not possible to wake up a communication node when this node has no power. In other words, the behavior of a communication systems in terms of start-up, shutdown, or wake-up depends on the basic principles of the car's power distribution.

This tight coupling between power supply and communication systems is also valid in terms of EMC. It is a common misunderstanding that optical communications system like MOST preclude EMC issues. However, also ECUs connected to the IVN by an optical communication system need power supply and EME is often generated via the power supply system. As addressed in the previous Section 4.4.1 the power supply systems are generally not designed to be symmetric in ground and plus. This can directly be measured in unfavorably EME values. Note that communication systems using coaxial cables face similar issues, as coaxial cables are asymmetric by definition. Transient currents can occur on the shield as a result of this asymmetry and the difficulties to correctly bond the shield to, e.g., the ground of the car's body. These transient currents are very

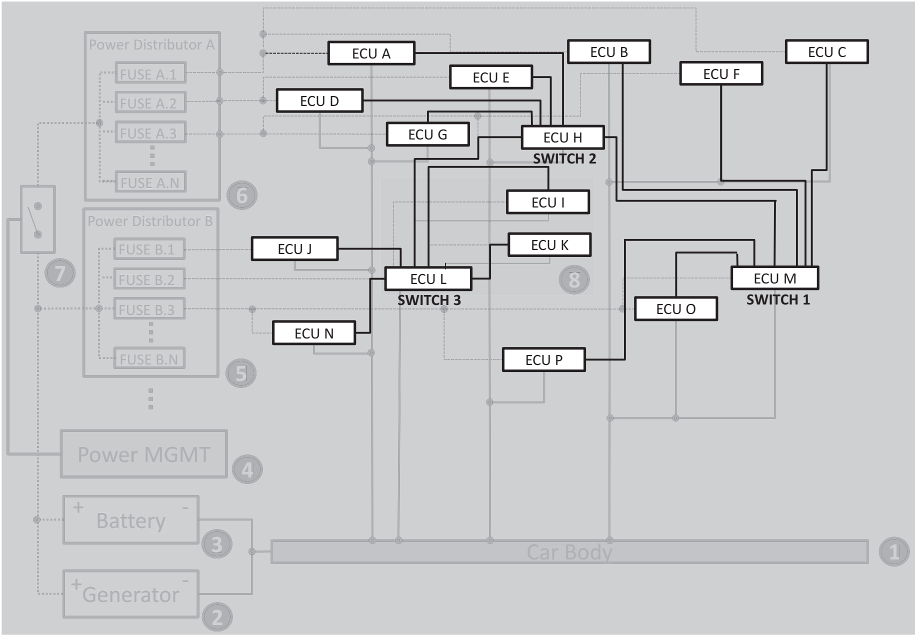

Figure 4.40 Example of the interrelation between the power distribution concept and an Ethernet communication network.

critical for the common mode noise. In order to suppress any impact the transient currents can have into the system, additional ferrite beads are used added to fix this issue.

4.4.3 Power over Data Line (PoDL)

In the 1990s, the assumption in automotive was that high-speed data transmission in a car required optical transmission. When using Plastic Optical Fiber (POF) for data transmission with solutions like MOST, there is no possibility to transport energy over the POF. An additional connector is required for the power supply of an ECU. Transmitting power with the data thus moved out of focus. This has dramatically changed in recent years, in which high-speed data transmission has gone electrical again. With using an electrical system for the high-speed data transmission, like, e.g., 100BASE-T1 Ethernet, it is again possible to additionally transmit power over the communication wires and designers will carefully consider its pros and cons.[17]

The principle behind PoDL is comparably simple, because the data is transmitted differentially, which for 100BASE-T1 this is achieved with a capacitor that blocks the DC part of the current. To transmit power as well, the common current is coupled onto the cables at the transmit side with help of a suitable circuit and coupled out again at the receive side. For two pair solutions like 100BASE-TX, the IEEE standardized the first version of "Power over Ethernet" (PoE) in 2003. For transmitting power over an Ethernet technology that uses one pair only, the respective task force was approved in November 2013 [97] and the PoDL standard IEEE 802.3bu was finalized in 2016 [98]. Some principle considerations are described in the following.

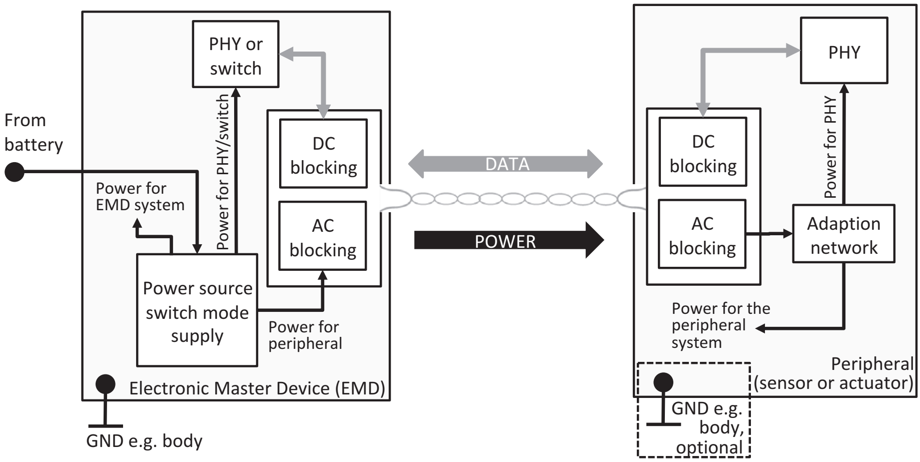

Figure 4.41 Example implementation for PoDL in cars.

As has been mentioned, the harness is an expensive component. So, especially for sensors that are located at the far ends of a car it is of particular interest to be able to use PoDL: The required power is generally low and the saving in cabling and connectors noteworthy. Reduction of weight or the need to reduce the diameter of the harness in small openings can be other reasons to deploy PoDL. It is even possible to control the voltage drop on a link such that there always is the optimal voltage available at the peripheral sensor, albeit simple applications might live with power directly from the general power supply system. The voltage drop will be limited according to the wire gauge and power consumption of the peripheral. The necessary electronic components need to be selected according to the needed current and in the simplest case consist of an inductivity. Unfortunately though, the inductivities need additional space, which in some use cases is not available.

A big advantage of PoDL is that it can further optimize the EME and EMI. The power supply wires are always also source of electromagnetic interference (conducted coupling). With the right termination and a good balance of the data wires, the current flow can be closed loop right back to the source. In this case only one ground level is used and ground shifts between different locations in the car are avoided. The full potential in Automotive Ethernet thus has not yet been exploited. Figure 4.41 shows the principle setup of PoDL with Automotive Ethernet.

4.4.4 Data over the Power Supply Network

When looking at the power supply network as described in Section 4.4.1 it is an obvious temptation to use the supply network also for data communication. After all, all ECUs need power so some wires have to reach all ECUs in a car. In nonautomotive applications Power Line Communication solutions are quite common, e.g., for realizing home networks without having to install extra wiring (see, e.g., [71]. In the context of vehicles PLC is important in the context of charging electrical vehicles, where additional data on vehicle status, ownership etc. is transmitted over the charging cable for billing purposes etc. (see, e.g., [99]). However, this communication is not so much inside the car, but for the communication to/with the outside. Because of the appeal of the idea, many investigations have been performed to enable PLC also for communication between ECUs inside the car (see, e.g., [100]). But because of the described unsymmetrical design of the power supply system and additional problems with special power supply filters, PLC is not (yet) economical inside the car and has, to the authors' knowledge, not been attempted for Automotive Ethernet.

4.4.5 Using Energy-Efficient Ethernet (EEE) in Cars

In automotive, it is an important requirement to be able to use Ethernet or any other IVN technology with low power consumption. This is true for the time an ECU and its communication are active, but it is even truer, when an ECU is idle and not needed for a while. The lowest power consumption is achieved when an ECU currently not needed is taken of the power supply (see also Sections 6.3.1 and 6.3.3), but also partial deactivation of ECUs is of interest. For CAN the low energy requirement resulted in its own standardization effort [101] and transceiver products. The question is thus, whether a similar activity is necessary for Automotive Ethernet or whether reuse from the existing IEEE efforts is possible.

In IEEE, the need to save energy has also been identified and addressed. With the introduction of switched Ethernet networks, units having no data to send, transmit idle symbols, in order to be able to constantly track the transmission conditions and to keep the synchronization. Naturally, this significantly increased the power usage, as suddenly all Ethernet nodes connected to a network, constantly transmit. In 2010 the IEEE thus published the IEEE 802.3az standard on Energy-Efficient Ethernet (EEE) [47].[18] The key element of EEE is the introduction of a Low-Power Idle (LPI) mode. When a link is idle and the two attached units would thus observe the traffic of idle packets only, the units can agree to go into LPI instead. During the LPI every 20 ms of zero transmission is followed by a 200 μs active refresh in order to keep the system synchronized.

This is, in principle, an appealing concept. The power needed to transmit idle symbols can be reduced in case the units currently have no data to transmit and there is thus no activity on the link; the target for EEE was to reduce power consumption of the PHY in the range of 50% [102]. Because of the regular refresh, a link can be reactivated fast. Maximum 16.5 μs start-up time were specified for 100 Mbps Ethernet and 30 μs for 1 Gbps Ethernet. For the use in automotive there are two aspects to consider:

1 An ECU consists of significantly more elements than just the PHY. To save 50% energy in the PHY in case of inactivity is a start, but not nearly enough from an automotive perspective. If an ECU is currently not needed or inactive, more energy should be saved.
2 The automotive use case is different from the one in the IT world. In an in-vehicle network, there are clear rules when an ECU is needed and when it is not needed, while user triggered activity in an IT network is less regular and predictable. To support short and irregular on–off phases as EEE does, is a huge advantage in IT.

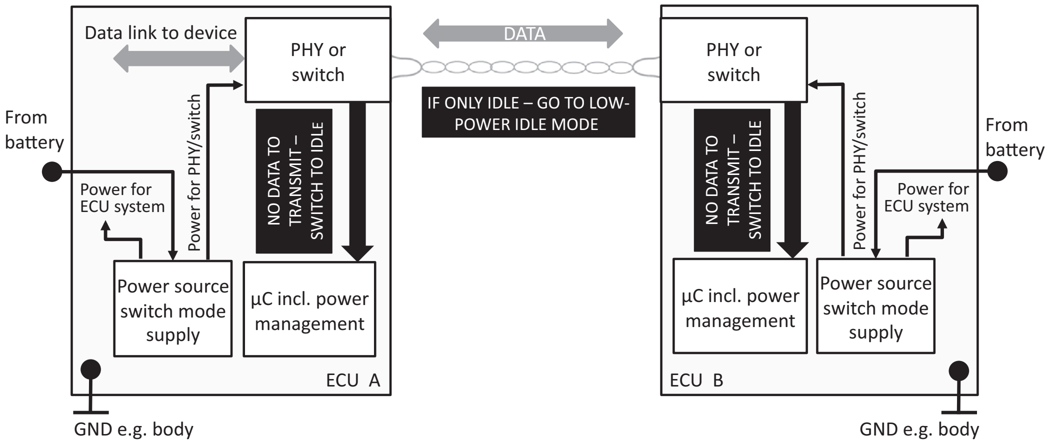

Figure 4.42 Example implementation for using power-efficient Ethernet in cars.

> For automotive use, EEE can also be deployed, but because of the different use patterns there are other methods that promise to save more power (e.g., see Section 4.4.6).

For RTPGE/1000BASE-T1, EEE is thus only an optional feature [12] (see also Section 4.3.2.1). The conclusion is that the right energy saving concept for ECUs connected via Automotive Ethernet yet needs to be developed. A first solution that takes care about the additional needs and restrictions in the automotive use cases is specified with the wake-up discussed in the next section. Figure 4.42 shows an example block diagram of how EEE would be realized in an automotive implementation.

4.4.6 Wake-Up

The OPEN Alliance developed a specification for wake-up [103] – to start with for 100BASE-T1, but with the idea to be extendable to 1000BASE-T1 – in order to close the gap between what can be achieved by EEE as described in Section 4.4.5 and a relatively easily manageable solution for in-car applications with maximum power efficiency. Maximum power efficiency means that not only the PHY, but the complete ECU is put to sleep and woken up only if needed again. This improves the lifetime of ECUs, reduces CO_2 emissions and increases the operating reach of electrical vehicles.

This chapter explains the basic principles behind the wake-up mechanism specified. Because the Ethernet communication system represents a switched network, it is one of the main challenges to be able to transport a wake-up request via the switches. At the time of writing, the BMW implementations still used additional wake-up lines to transport the wake-up pattern to all respective ECUs connected to the network. With this new specification the wake-up can be achieved without the wake-up line.

One of the most critical moments in automotive applications is the start-up/shutdown procedure. One of the goals of the OPEN Alliance specification therefore was to keep the solution as simple as possible. This includes, that the principle behind the start-up/shutdown procedure defined and described below, follows the same principle as is used with a wake-up line.

The main objectives behind the specification were [104]:

- CO_2 savings can be achieved without affecting any performance or driving experience
- All vehicle functions remain available at any time
- No modification of existing network architecture, no additional components are required
- Wake-up process completely covered in ISO/OSI layer 1
- Global network wake-up (incl. link start-up time) within less 250 ms
- Controlled link shutdown to deactivate selective parts of network
- No unwanted wake-up in presence of interference noise
- Applicable for 1000BASE-T1 and 100BASE-T1

Therefore, the specification covers the following basic functions:

- Wake-up reception and signalizing
- Wake-up transmission
- Wake-up forwarding
- Sleep

Fundamental requirements to utilize this method are of course that various vendors offer components that support this method and that the power supply is not switched off centrally. The ECUs with Ethernet switched or at least the switch semiconductor inside respective ECUs needs to have power for this to work. This would be easier to ensure if semiconductor controlled power distribution was used that allows for more individualized power control. With the OPEN Alliance wake-up ECUs in a continuously supplied subsystems can be turned off, and with the appropriate commands again come to life. This is needed to have a cost effective and high energy-efficient system design.

The basic idea is – like with a separate wake-up line – that upon a single wake-up trigger all ECUs previously in sleep mode start-up, and those not needed simply go back to sleep as quickly as possible. The important advantage of this type of approach is that knowledge on the different use cases, i.e., what device is needed to fulfill a special functionality, has not to be inherited to the lower layers of the communication stack. The basic principle of this wake-up method is depicted in Figure 4.43.

At first sight, it seems disadvantageous that all units need to wake up upon a single wake-up impulse in the system, even if many of them are not needed and can/will go back to sleep. However, to proceed such ensures that all decisions are taken on the ISO/OSI layers foreseen for this decision. It is the application that knows whether an ECU is needed for a specific function or not. To alternatively use a wake-up/sleep method that included such knowledge on lower layer, would mean to break with the ISO/OSI layering and could potentially result in an avalanche of unwanted interdependencies that jeopardizes the complete system design.

To allow for a better understanding of the system behavior, Figure 4.44 depicts a start-up in the example system used in this section. The left upper diagram (1) shows the initial state in which only ECUs I, L, and N are active and communicating. The upper

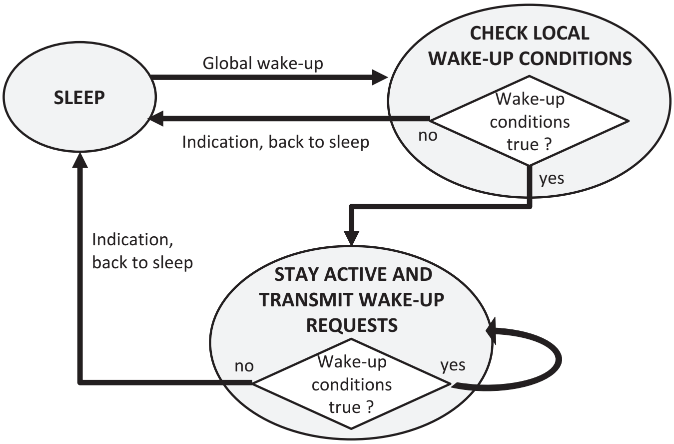

Figure 4.43 The basic state machine for the OPEN Alliance wake-up/sleep.

right diagram (2) shows a wake-up request triggered inside the function of the ECU C, which is then forwarded into the network (3). As switch 1 is the units to first receive this request it forwards it via all its ports (4). One of the units that receives the wake-up request from switch 1 is switch 2 integrated in ECU H. This switch again forwards the wake-up request through all its ports (5). The next switch to receive the wake-up request is switch 3. Switch 3 will forward the wake-up request only on the links not yet active (6). Links that have been active all along are thus not affected by this forwarding action.

At this point in the sequence the whole system is awake (6). Now, that the whole system is activated, each node decides via higher layers in the communication stack if its functionality is needed in the use case that triggered the wake-up (7). If this is not the case, the units decides to shut down following their specific shut down sequence. This means that nodes not needed go back to sleep and the sequence ends in the next stable state (8). In the example of Figure 4.44 the units having gone back to sleep are A, B, E, F, G, K, O, and P.

4.5 The Quality Strain

Addressing quality in cars means considering the complete production chain. It starts with the IC silicon design, includes PCB design on component-level, ECU design, the manufacturing process at Tier 1, and the manufacturing process of the car at the OEM. The quality has to last for a minimum of ten years of customer use and for use in a very challenging physical environment that faces extreme temperature variations, very wet as well as very dusty environments, vibrations, dirt, and electrical stress to name a few. In order to handle this, clear processes are needed for every aspect, which altogether make the "Automotive Quality." As this book is about Automotive Ethernet, this section will focus on the quality of the active semiconductors and passive components needed for deploying the networking technology.

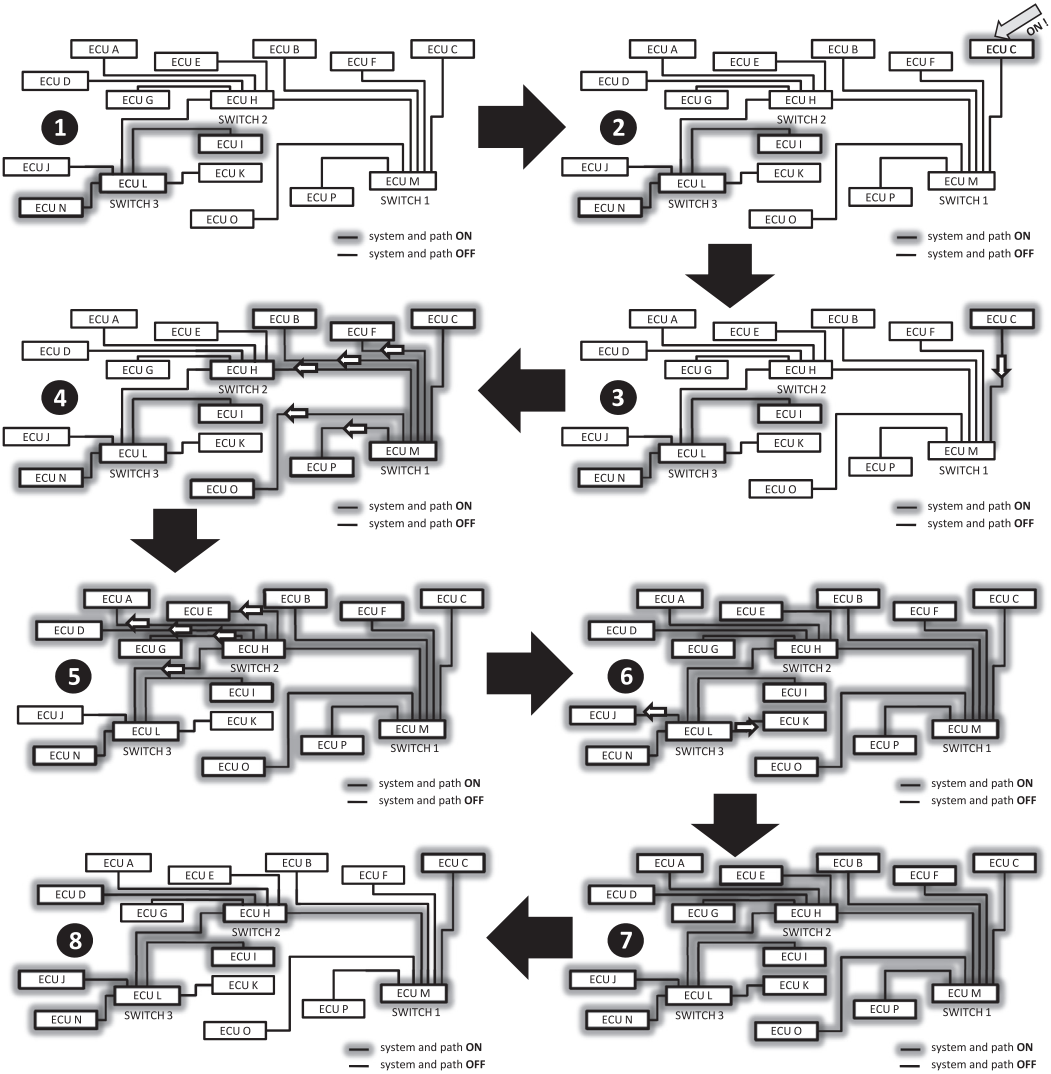

Figure 4.44 Start-up sequence following a wake-up request.

4.5.1 Automotive Semiconductor Quality Standards

A common quality value well suited to visualize the difference between automotive and other industries is the statistic defect rate for one Million parts, indicated in Parts Per Million (PPM). Naturally, zero defects are the ideal target value. But, the lower the PPM value the more expensive the part will be, which means that every industry and inside every industry every company has to find its own compromise between the price a company is willing to pay and the target quality it wants to achieve.

Imagine a car manufacturer who builds one Million cars a year. Every car has 100 ECUs and every ECU consists of 400 electronic parts. This means 40.000 electronic

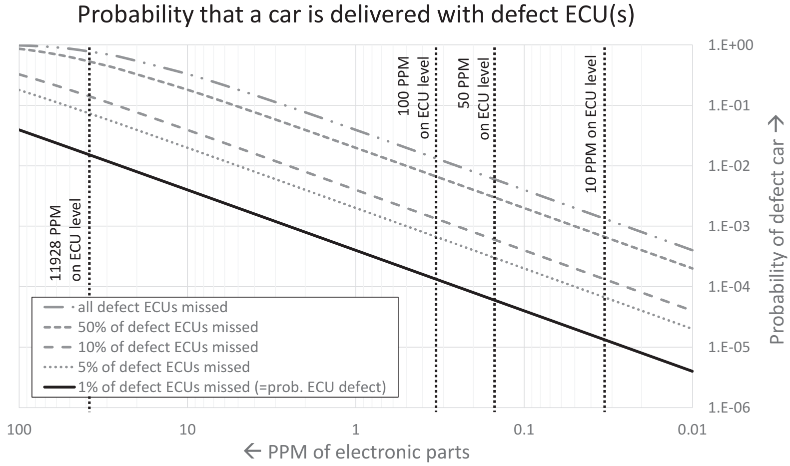

Figure 4.45 Relationship between PPM and percentage of defect cars, depending on the percentage of defect ECUs not detected ("missed") during the qualification process. The assumption is that every car consists of 100 ECUs and that every ECU consists of 400 parts. Today, not many cars have this complexity, but it shows that the more complex a car is and the more electronics a car contains, the more critical is the quality of every component.

parts per car, and 40 Billion parts the car manufacturer build into cars per year. Note that 400 electronic parts per ECU, does not represent the average of today, but it represents a simplified future into which the car industry is heading. Figure 4.45 shows the probability of having a defect ECU and the probability of having delivered a car with a defect ECU to a customer, both depending on the PPM on part level.[19] The additional parameter is the percentage of defect parts that are not or cannot be detected in an ECU at the time of production.

At 30 PPM on electronic part level – a very good quality value in the consumer industry [105] – a car manufacturer would deliver about 70% of cars with at least one defect ECU, were none of these faulty ECUs detected in time in the production chain. The vertical lines in Figure 4.45 indicate the respective PPM values on ECU level. It can be seen that 100 PPM on ECU level mean about 0.25 (!) PPM on part level. This would still result in every 100th car containing a defect ECU if all defects were missed and every 1000th car if 10% of the defect ECUs were missed; quality values completely unacceptable from a customer's perspective. The solution, of course, would be to simply find all defect ECUs before they are built into the car. Unfortunately, even with very good final qualification methods, this is not possible, as the malfunctions in defect semiconductors often only materialize in the one too many borderline situations of extreme temperature, humidity, or mechanical stress. The solution is thus to improve the production process of semiconductors further and to standardize the qualification of the semiconductors such that not only faulty semiconductors but also those more likely to be faulty at a later point in time are not used in products in the first place.

At the beginning, when electronics started to become more frequent in the automotive industry every car manufacturer had its own requirements and every supplier its own qualification plan that in return needed to be reviewed by the car manufacturer. So, in

Table 4.12 Overview of AEC specifications on quality test methods for integrated circuits AEC-Q100

Standard	Content
AEC – Q100	Stress qualification for integrated circuits (base document only with no test methods)
AEC – Q100–001	Wire bond shear test
AEC – Q100–002	Human Body Model (HBM) electrostatic discharge test
AEC – Q100–003	Machine Model (MM) electrostatic discharge test
AEC – Q100–004	IC latch-up test
AEC – Q100–005	Nonvolatile memory program, erase endurance, data retention, and operational life test
AEC – Q100–006	Electrothermally induced parasitic gate leakage test
AEC – Q100–007	Fault simulation and test grading
AEC – Q100–008	Early Life Failure Rate (ELFR)
AEC – Q100–009	Electrical distribution assessment
AEC – Q100–010	Solder ball shear test
AEC – Q100–011	Charged Device Model (CDM) electrostatic discharge test
AEC – Q100–012	Short circuit reliability characterization of smart power devices for 12 V systems

1992 US car makers came up with the idea that lead to the Automotive Electronics Council (AEC), which then standardized quality in the yet small but increasing market of automotive semiconductors [106]. The initial standard developed by the AEC was the AEC-Q100 for Integrated Circuits (ICs). After having been reviewed by primary IC suppliers, it was available in June 1994. All three US car makers Chrysler, Ford, and GM accepted only AEC-Q100 qualified semiconductors and thus achieved a major milestone in standardized automotive quality [106].

Today the AEC-Q100 is a minimum standard for automobile manufacturers and suppliers worldwide. Later specifications for two more categories of semiconductors were added: AEC-Q101 for discrete semiconductor devices and AEC-Q200 for passive components. Up to this very day, the AEC still organizes annual reliability workshops [107].

To give an overview and an idea about the complexity of the quality requirements Table 4.12, Table 4.13, and Table 4.14 list the respective specifications.

The AEC quality specifications are important for Automotive Quality, albeit not comprehensive. Car manufacturers often have additional requirements and work in close relation with key automotive semiconductor manufacturers with respect to quality management, design rules, test coverage, test strategy, and process technologies for the

Table 4.13 Overview of AEC specifications on quality test methods for discrete semiconductors AEC-Q101

Standard	Content
AEC – Q101	Stress test qualification for discrete semiconductors (base document only, no test methods)
AEC – Q101–001	Human Body Model (HBM) electrostatic discharge test
AEC – Q101–002	Machine Model (MM) electrostatic discharge test
AEC – Q101–003	Wire bond shear test
AEC – Q101–004	Miscellaneous test methods
AEC – Q101–005	Charged Device Model (CDM) electrostatic discharge test
AEC – Q101–006	Short circuit reliability characterization of smart power devices for 12 V systems

Table 4.14 Overview of AEC specifications on quality test methods for passive components AEC-Q200

Standard	Content
AEC – Q200	Stress test qualification for passive components (complete document with test methods)
AEC – Q200–001	Flame retardance test
AEC – Q200–002	Human Body Model (HBM) electrostatic discharge test
AEC – Q200–003	Beam load (break strength) test
AEC – Q200–004	Measurement procedures for resettable fuses
AEC – Q200–005	Board flex / terminal bond strength test
AEC – Q200–006	Terminal strength / shear stress test
AEC – Q200–007	Voltage surge test

different stages of development, ramp up, and serial production. On ECU level additional requirements exists and they might vary depending, e.g., on the use of an ECU in a so-called "wet area" or "dry area" or in case additional temperature requirements need to be fulfilled. These additional requirements often vary among car manufacturers, which, in the end, also reflects the quality a customer can expect in his/her car.

4.5.2 The CMC (Quality) for Automotive Ethernet

While the previous section gave an idea of the complexity of (semiconductor) quality in automotive in general, this section gives an example for the impact the automotive quality requirements can have on the actual use of a technology. In case of Automotive Ethernet, the quality of the transceiver (PHY) semiconductor was less critical. If a semiconductor company cannot afford to setup production according to the automotive requirements in-house, foundries like the Taiwan Semiconductor Manufacturing Company (TSMC) can perform the automotive qualified production on their behalf. Basic automotive quality can thus be achieved with reasonable effort.[20]

However, Ethernet-based communication does not only require a suitable PHY chip but also coupling to the network. For this, transformers are used. Transformers provide DC blocking in form of a galvanic separation between PCB and transmission line and they perform a common mode rejection/suppression in order to optimize the design and to comply with EMC requirements. Figure 4.46 shows the schematic and x-ray of a standard 100BASE-TX transformer. The inductor cores are generally wound and soldered by hand before being cast into the housing. Figure 4.46 shows the irregular winding and how the cores are placed into the housing without any additional fixing. In principle,

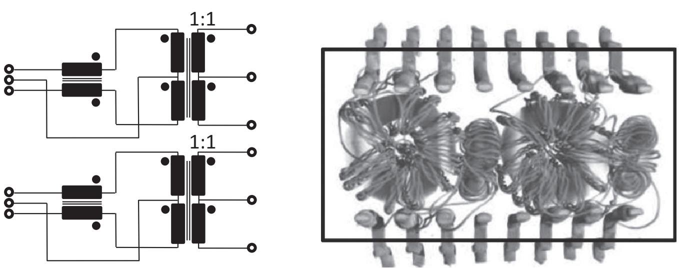

Figure 4.46 Schematic and x-ray of a typical 100BASE-TX transformer.

the wires are isolated. However, when the housing is soldered onto the PCB it is possible that some of the wires running unintentionally close to other pins than they are connected to, are fixated during the soldering process at a few more places than desired.

The variation in quality of such produced parts tends to be high and is thus not acceptable for the automotive industry. Any handmade component is critical. This means that in order to be able to use Ethernet in automotive, another solution was needed. The production of "planar transformers" can be automated, but they are not (yet) competitive in terms of costs and size.[21] The first step therefore is to look at the need for transformers as such. In cars, the high insulation voltage standard Ethernet transformers have to deal with, is not needed. Cars use 12 V only and have a common ground. This allows realizing the DC isolation with capacitors instead of using the transformer part of the schematic shown in Figure 4.46. However, common mode suppression is vital for the Electro Magnetic Immunity (EMI). This function does need to be provided for, which can be done by a Common Mode Choke (CMC).

For 100BASE-T1/OABR, BMW first intended to use a ring core CMC. This reduced the complexity in comparison with the transformer shown in Figure 4.46. The challenge, however, was to fully automate the production of the CMC. It turned out not to be possible. In order to meet the automotive quality requirements nevertheless, this would have meant to test every single produced part individually. As a result the CMC would have been more expensive than the PHY IC. This was not acceptable, but is a good example on how a quality requirement can potentially prohibit the use of a technology.

A different solution was needed. After some intensive research and development work an I-core variant was selected for the CMC. This variant allows fully automated production and the expected small variance in quality, which can be achieved with the CMCs used for other in-vehicle networking technologies like CAN and FlexRay. In the end, only the discovery of this CMC variant made Ethernet-based communication cost competitive in automotive use.

When a final, working solution is available, it is no longer possible to deduct the effort to achieve it. Critics of Automotive Ethernet always suspected the complementary hardware to be the cost driver of Ethernet that would, in the end, make it too expensive. Without the usability of an I-core CMC, they might have been right. This example shows, how important it is to always look at the complete system. It also explains why the automotive industry needs significant lead time to introduce a new technology.

Notes

1 EMI is sometimes also used as an abbreviation for ElectroMagnetic Interference, which might be used as a general term for EMC behavior. Using EME and EMI as defined in this book seems more unambiguous. EME is how you might disturb others, EMI is how you are being interfered.

2 The measured emissions are somewhat closer to the limit line than was the case for the first ever automotive 100BASE-T1/OABR measurement shown in Figure 3.6. This is due to the fact that the transmit power was increased for these later measurements. As 100BASE-T1/OABR had sufficient margin to the limit line, transmit power was added in order to further improve the immunity.

3 As not all car manufacturers have published their limit lines, it is not possible for the authors to claim more than "many."
4 In cars, vans, and light trucks, 15 m has been identified as the maximum length. The original BroadR-Reach specification even foresaw a link length of 10 m only, which for passenger cars is more than sufficient [108]. In contrast, in busses or large trucks a link length of 40 m is required [32].
5 At IEEE 802.3 these channels were mainly specified at the TIA, sometimes at the same time as the IEEE Ethernet technology. The table below shows the timeline.

Ethernet name	IEEE number	Year of publication	TIA channel used	Year of publication
10BASE-T	802.3i	1990	CAT 3 (telephone wiring)	1991
100BASE-TX	802.3u	1995	CAT 5	1995
1000BASE-T	802.3ab	1999	CAT 5e	1999
10GBASE-T	802.3an	2006	CAT 6 (shorter distances), CAT 6A	2002, 2007
40GBASE-T	802.3bq	ongoing	CAT 7A	2009

6 According to IEEE, 100BASE-TX can also be used in "full-duplex" mode, as the definition of full or half-duplex at IEEE is not based on what happens on the channel, but on what happens at the (x)MII interface. Thus only when the media is shared, like when using the CSMA/CD mode, is this referred to as "half-duplex," while full duplex indicates the use of a switched network. With each pair of wires in 100BASE-TX either being able to transmit or receive only, the authors consider 100BASE-TX as "half-duplex" on channel level, even if the meaning is different from CSMA/CD. In order not to cause too much naming confusing, this book will use "dual simplex" for 100BASE-TX. The occasional addition "true" to full duplex, results from trying to be unambiguous in the naming and might be used when transmit and receive signal simultaneously really use the same cable pair(s).
7 Not every LFSR results in a sequence of maximum length that is repeated after $2^n - 1$ register shifts. A suitable LFSR should have few exor functions, should be long enough to be able to create an almost (pseudo) random sequence and should have the right length for the transmission power to be relatively evenly spread over the power density spectrum. LFSRs that do cause a repetition after $2^n - 1$ register shifts use so-called "primitive" polynomials [109] [110] [111].
8 During transmission, the signals on the line generally get distorted, in amplitude and in shape. This means that the PAM3 symbol "0" is transmitted as 0 V, it might actually be received as 0.1 V = "0" if the disturbance is low, or as 0.6 V = "1" if the disturbance is high. While the Symbol Error Rate (SER) is independent of the 3B2T encoding used but solely depends on the noise added, it makes a difference for the BER. With a Gray code, detecting a "1" instead of a "0" for one of the PAM3 symbols in the 2D-PAM3 conversion, this will result in one bit error. In the 3B2T coding of 100BASE_T1 used, it might result in two bit errors.
9 Note that this number changed from the BroadR-Reach to the 100BASE-T1 specification. For BroadR-Reach a much longer maxwait_time of 1406ms ± 18ms for the master and 656ms ± 9ms for the slave had been defined [108].
10 The link_control parameter is normally used by the autonegotiation function to overrule other system functions in case of required changes initiated by it. In case autonegotiation is not activated/foreseen in the system (like it is the case for 100BASE-T1), the link_control parameter can be expected to be enabled continuously as soon as the system has power.
11 As was described in Section 3.3, it had been BMW's first attempt to create a 100BASE-TX based solution that passes the automotive requirements. In 2007 this was not successful. The authors therefore expect that the general learning on EMC for 100 Mbps Ethernet that have

been generated in order to enable 100BASE-T1/BroadR-Reach for automotive and technological progress also helped to find more suitable measures to improve the EMC behavior of 100BASE-TX hardware.

12 Blue, green, and yellow light actually show better attenuation behavior for POF than red light [112]. However, blue light only recently started to get commercially exploited (e.g., Blu-ray) and more uses might appear in the future.

13 The technology was first published at the VDE/DKE v 0885–763 in 2012 [113] before its standardization was launched at IEEE 802.3 with a CFI in March 2014 and the respective IEEE802.3bv alias 1000BASE-RH TF in January 2015.

14 As discussed in Section 2.2.2, CAN FD is designed for shared gross data rates of 2 Mbps and 5 Mbps. As the technology is very new, published data on in-vehicle implementation experience are yet rare. However, the proposed maximum load for a CAN link is 50% only, so it can be expected that also for CAN FD the proposed data rate will be significantly lower than 2 Mbps, 5 Mbps, and especially 10 Mbps. The same for FlexRay (see Section 2.2.5), which has a gross data rate of 10 Mbps. The experience at BMW showed that the system overhead as well as the actually chosen segmentation between static and dynamic section, can reduce the effective data rate easily to 20% (i.e., to 2 Mbps).

15 An important milestone was set with the so-called Paris agreement concluded by the United Nations Framework Convention on Climate Change (UNFCCC) in December 2015. Its goal is limit the global warming to below 2°C (ideally to 1.5°C) above preindustrial level [114]. Prior to the meeting, every country was asked to submit commitments on how to reduce greenhouse gas and carbon output in their countries. Based on 146 submittals, it was concluded that the proposed methods are not sufficient to achieve the target, but that even more stringent actions are needed [115].

16 The internal investigations distinguished between the supply network, the communication network and discrete wiring. In-vehicle networks were introduced, because discrete wiring would have been unsustainable with the increasing complexity of the EE architecture (see also Section 2.1). However, using a networking technology is not always efficient, nor is it always possible or sometimes simply has not been done yet. In consequence, the discrete wires still amount to the second largest share in the weight of a typical wiring harness.

17 For the latest developments of other communication technologies like pixel links (see Section 2.2.6) or the MOST cPHY (see Section 2.2.4.2), which use coaxial cabling for digital data transmission, the possibility to be able to transmit power with the data is an economic must. Systems using STP cabling generally allow to include a pair of wires within the cables for power. However, with coaxial cabling this is not possible and not only an additional wire pair but a completely different connector would be needed for the power supply, which in many cases makes such solutions economically unattractive. Because of the asymmetry of the coaxial cables, special attention has to be paid to the EMC effects when power is transmitted additionally. For UTP neither issue exists. A separate power supply can simply use separate pins on the same connector and PoDL finds a very symmetric cable.

18 Before EEE was finalized, apparently various companies already sold Ethernet equipment, which allowed reducing the power consumption. These methods were marketed under the name “Green Ethernet” [116] [117].

19 Assuming that the probability of a defect is the same for all electronic parts used, the probability to have selected X defect parts in an ECU is

$$\text{Prob}X\text{ Defects in ECU} = \frac{\text{No Parts ECU!}}{X!\,(\text{No Parts ECU} - X)!}\text{Prob Defect Part}^{X}$$
$$(1 - \text{Prob Defect Part})^{\text{No Parts ECU}-X}$$

which means that the probability to have at least one defect in an ECU is

$$\text{Prob ECU Defect} = \text{Prob Min1Defect in ECU} = 1 - \text{Prob 0 Defects in ECU}$$
$$= 1 - (1 - \text{Prob Defect Part})^{\text{No Parts ECU}}$$

Assuming that y% of defect ECUs are not detected before the ECU is built into the car, the probability to have an undetected defect and at a later point in time malfunctioning ECU in the car can be calculated similarly:

$$\text{Prob Car with Defect ECU} = \text{Prob Min 1 ECU Defect} = 1 - \text{Prob 0 ECU Defect}$$
$$= 1 - (1 - \text{y Prob ECU Defect})^{\text{No ECUs}}$$

with Prob Defect Part $= \frac{PPM}{1.0\langle/i\rangle 00.000}$. In the example No Parts ECU = 400 and No ECUs = 100 .

20 Better automotive quality requires effort and know-how that impacts the development very early in the design process.

21 Planar transformers can sometimes be found in the consumer and IT industries, especially in case of high frequency applications. As their production is fully automated they achieve better quality values than standard transformers. Planar transformers are directly integrated as part of the PCB and encased with ferrite material [49]. Apart from the automatable production, planar magnetics have performance advantages in low profile structure, low leakage current, reduced high frequency winding loss and better thermal management. On the downside they have a low window utilization factor and an increased parasitic capacitance [118]. The authors guess that the reasons planar magnetics are not well established in the market are due to size and costs.

References

[1] IEC, "The Electromagnetic Environment," 2013. [Online]. Available: www.iec.ch/emc/explained/environment.htm. [Accessed 26 December 2013].

[2] D. E. Möhr, "Was ist eigentlich EMV? – Eine Definition," 2013. [Online]. Available: www.emtest.de/de/what_is/emv-emc-basics.php. [Accessed 17 December 2013].

[3] Wikipedia, "CISPR," 7 August 2016. [Online]. Available: http://de.wikipedia.org/wiki/CISPR. [Accessed 25 September 2016].

[4] Learn EMC, "Introduction to EMC," undated. [Online]. Available: www.learnemc.com/tutorials/Introduction_to_EMC/Introduction.html. [Accessed 26 December 2013].

[5] Williamson Labs, "EMC," 1999–2011. [Online]. Available: www.williamson-labs.com/ltoc/glencoe-emc-11.htm. [Accessed 26 December 2013].

[6] R. Schmitt, *Electromagnetics explained*, Amsterdam: Newnes, 2002.

[7] K. Budweiser, "A theoretical Analysis of EMC Influence on Advanced Ethernet," Technische Universität München, Munich, 2013.

[8] B. Körber, D. Sperling, and T. Form, "EMV Bewertung von CAN transceivern," 2001. [Online]. Available: www.fh-zwickau.de/fileadmin/ugroups/ftz/Publikationen/Prof_Sperling/2001_EMV-Beurteilung.pdf. [Accessed 17 December 2013].

[9] M. Tazebay, *per e-mail*, 2013.

[10] B. Deutschmann, R. Jungreithmair, and G. Winkler, "Messmethode zur Messung der leitungsgeführten Störemissionen von ICs," 2003. [Online]. Available: www.eue24.net/pi/media/pdf/eue/emck_2003/ek3c0101.pdf. [Accessed 14 February 2014, no longer available].

[11] M. Tazebay, "A Modeling & Measurement Technique for RTPGE EMC Channel Analysis," 24 January 2013. [Online]. Available: www.ieee802.org/3/bp/public/jan13/tazebay_3bp_01a_0113.pdf. [Accessed 14 May 2016].

[12] IEEE-SA, "IEEE 802.3bp-2016 Amendment 4: Physical Layer Specifications and Management Parameters for 1 Gb/s Operation over a Single Twisted Pair Copper Cable," IEEE-SA, New York, 2016.

[13] M. Tazebay, "Immunity Analysis & Test Results via Bulk Current Injection Method for 1-Pair UTP Channels," 15 May 2013. [Online]. Available: www.ieee802.org/3/bp/public/may13/tazebay_3bp_01_0513.pdf. [Accessed 14 May 2016].

[14] A. Chini and M. Tazebay, "RTPGE BCI Noise Analysis for Common Mode Termination & Grounding Effects," 12 November 2013. [Online]. Available: www.ieee802.org/3/bp/public/nov13/Chini_Tazebay_3bp_02a_1113.pdf. [Accessed 14 May 2016].

[15] G. Zimmerman, "Salz SNR Text and Procedure," 25 August 2015. [Online]. Available: www.ieee802.org/3/NGEBASET/public/archadhoc/zimmerman_3bzah_01a_0815.pdf. [Accessed 14 May 2016].

[16] A. Vareljian and H. Takatori, "Channel Qualification Based on Salz," September 2012. [Online]. Available: www.ieee802.org/3/bj/public/sep12/vareljian_3bj_01_0912.pdf. [Accessed 14 May 2016].

[17] M. Tazebay, "A Review of 1000BASE-T1 PHY Architecture for EMC Constrained Channels," in *IEEE-SA (4th) Ethernet&IP@Automotive Technology Day*, Detroit, 2014.

[18] P. Leroux and M. Steyaert, *LNA-ESD Co-Design for Fully Integrated CMOS Wireless Receivers*, Cham: Springer (Kluwer), 2005.

[19] ESD Association, "Fundamentals of Electrostatic Discharge," 2010. [Online]. Available: www.esda.org/assets/Uploads/documents/FundamentalsPart5.pdf. [Accessed 25 September 2016].

[20] S. Carlson, K. Matheus, T. Hogenmüller, T. Streichert, D. Pannell, and A. Abaye, "Reduced Twisted Pair Gigabit Ethernet Call For Interest," 15 March 2012. [Online]. Available: www.ieee802.org/3/RTPGE/public/mar12/CFI_01_0312.pdf. [Accessed 16 March 2012].

[21] B. Körber, "BroadR-Reach, Definitions for Communication Channel, Version 2.0," OPEN Alliance, 2014.

[22] IEEE-SA, "IEEE 802.3bw-2015: Amendment 1 Physical Layer Specifications and Management Parameters for 100 Mb/s Operation over a Single Balanced Twisted Pair Cable (100BASE-T1)," IEEE-SA, New York, 2015.

[23] B. Körber, "BroadR-Reach, Definitions for Communication Channel, Version 1.0," OPEN Alliance, 2013.

[24] C. Arndt, "Determination of Relevant Parameters for the Diagnosis of BroadR-Reach: Prototypic Realization for the Diagnosis of the Communication Channel," Hochschule Deggendorf, Deggendorf, 2013.

[25] B. Körber, "IEEE 1000BASE-T1 EMC Test Specification for Common Mode Chokes," OPEN Alliance, Irvine, 2016.

[26] Wikipedia, "Scattering parameters," 25 September 2016. [Online]. Available: http://en.wikipedia.org/wiki/Scattering_parameters. [Accessed 25 September 2016].

[27] C. DiMinico, "802.3bp Cabling parameters to s-parameter naming," September 2014. [Online]. Available: www.ieee802.org/3/RTPGE/email/pdfDOLXj6Te3J.pdf. [Accessed 15 March 2016].

[28] B. Körber and T. Wunderlich, *per e-mail*, 2016.

[29] N. A. Wienckowski, "I Can Use CAN Wire for Ethernet, Right?," in *IEEE-SA (4th) Ethernet&IP@Automotive Technology Day*, Detroit, 2014.

[30] D. Gräber, Used from nonpublic document, Munich, 2013.

[31] K. Matheus et al., "1 pair or 2 pairs for RTPGE: Impact on System Other than the PHY Part 1: Weight & Space," 23 January 2013. [Online]. Available: http://grouper.ieee.org/groups/802/3/bp/public/jan13/matheus_3bp_01_0113.pdf. [Accessed 23 January 2013].

[32] S. Buntz, "IEEE RTPGE Information: Update on Required Cable Length," 17 July 2012. [Online]. Available: http://grouper.ieee.org/groups/802/3/RTPGE/public/july12/buntz_02_0712.pdf. [Accessed 17 July 2012].

[33] L. Winkel, "CFI 'Reduced Pair Gigabit Ethernet PHY' Industrial Environment," 26 April 2012. [Online]. Available: http://grouper.ieee.org/groups/802/3/RTPGE/public/may12/winkel_01_0512.pdf. [Accessed 28 December 2013].

[34] S. Carlson, "Reduced Twisted Pair Gigabit Ethernet Study Group: Objectives," 15 November 2012. [Online]. Available: www.ieee802.org/3/RTPGE/Approved_Objectives_1112.pdf. [Accessed 30 November 2012].

[35] C. DiMinico, "Reduced Twisted Pair Gigabit Ethernet Link Segment Definitions Decision Process," 22 March 2013. [Online]. Available: www.ieee802.org/3/bp/public/mar13/diminico_3bp_02_0313.pdf. [Accessed 29 December 2013].

[36] C. DiMinico and M. Tazebay, "IEEE P802.3bp (RTPGE) PHY Task Force Channel Definitions Ad Hoc Report," November 2013. [Online]. Available: www.ieee802.org/3/bp/public/nov13/ch_ad_hoc_3bp_01_1113.pdf. [Accessed 25 December 2013].

[37] M. Tazebay and S. Buntz, "Reduced Twisted Pair Gigabit Ethernet EMC & Noise Ad Hoc Report," 12 November 2013. [Online]. Available: www.ieee802.org/3/bp/public/nov13/EMCnoise_ad_hoc_3bp_01_1113.pdf. [Accessed 25 December 2013].

[38] W. Larsen, R. Mei, B. Moffitt, and T. Herman, "RTPGE Channel Requirements Proposal for 1-Pair Ethernet," 10 July 2013. [Online]. Available: www.ieee802.org/3/bp/public/jul13/herman_3bp_01_0713.pdf. [Accessed 29 December 2013].

[39] T. Müller and B. Bergner, "Channel Component Requirements for 1000BASE-T1 Automotive Ethernet, v. 1.1," OPEN Alliance, Irvine, 2016.

[40] Broadcom, "YD/T 1947–2009: Technical Requirement of Physical Layer for Extended Reach Ethernet based on 2D-PAM3 and 4-D PAM5," Ministry of Industry and Information Technology, PRC, Beijing, 2009.

[41] C. Mick, C. DiMinico, S. Raghavan, S. Rao, and M. Hatamian, "802ab, a Tutorial Presentation," 17 March 1998. [Online]. Available: http://grouper.ieee.org/groups/802/3/tutorial/march98/mick_170398.pdf. [Accessed 27 November 2013].

[42] IEEE-SA, "IEEE 802.3–2008 Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications," IEEE-SA, New York, 2008.

[43] F.-J. Kauffels, "1000 BASE-T: Gigabit Ethernet über Kupferverkabelung (1/2)," 18 September 2012. [Online]. Available: www.comconsult-research.de/1000-base-t-gigabit-ethernet-uber-kupferverkabelung-12./. [Accessed 27 February 2014].

[44] S. Patwardhan, "Gigabit Ethernet over Copper: Hardware Architecture and Operation," 2001. [Online]. Available: www.dell.com/content/topics/global.aspx/power/en/ps4q01_patward?c=u. [Accessed 13 November 2016].

[45] W. Buchanan, "Gigabit Ethernet Over Category 5 U.T.P. Cabling," Master's Thesis, undated.

[46] B. Noseworthy, "Gigabit Ethernet 1000BASE-T," 29 March 2000. [Online]. Available: www.iol.unh.edu/services/testing/ge/knowledgebase/pcs.pdf. [Accessed 26 December 2013].

[47] IEEE-SA, "802.3az-2010 Amendment 5: Media Access Control Parameters, Physical Layers, and Management Parameters for Energy-Efficient Ethernet," IEEE-SA, New York, 2010.

[48] O. E. Agazzi, J. L. Creigh, M. Hatamian, D. E. Kruse, A. Abnous, and H. Samueli, "Multi-Pair Gigabit Ethernet Transceiver," US Patent 2011/0064123 A1, 17 March 2011.

[49] Ethernet Alliance, "Ethernet Alliance Panel #2: Bandwidth Growth and the Next Speed of Ethernet," 2012. [Online]. Available: www.ethernetalliance.org/wp-content/uploads/2012/04/Ethernetnet-Alliance-ECOC-2012-Panel-2.pdf. [Accessed 15 March 2013].

[50] M. Tazebay and T. Suermann, "Deployed 100Mb/s One Pair OABR PHY," May 2012. [Online]. Available: www.ieee802.org/3/1TPCESG/public/tazebay_01_0512.pdf. [Accessed 5 November 2016].

[51] IEEE 802.3, "IEEE 802.3ba 40Gb/s and 100Gb/s Ethernet Task Force Homepage," 19 June 2010 (closed). [Online]. Available: http://grouper.ieee.org/groups/802/3/ba/index.html. [Accessed 25 December 2013].

[52] A. Abaye, "BroadR-Reach Technology: Enabling One Pair Ethernet," 2013. [Online]. Available: www.ethercat.org/2013/mobile_applications/files/04_EtherCAT_Mobile_App_Broadcom.pdf. [Accessed 27 November 2013].

[53] Wikipedia, "Gray Code," 22 October 2016. [Online]. Available: https://en.wikipedia.org/wiki/Gray_code. [Accessed 27 October 2016].

[54] Züricher Hochschule Winterhur, "Kapitel 5: Digitale Übertragung im Basisband," 2005. [Online]. Available: https://home.zhaw.ch/~rur/ntm/unterlagen/ntmkap51einbb.pdf. [Accessed 5 November 2016].

[55] M. Jones, "Zwickau University of Applied Science EMC Lab Testing for Automotive Ethernet Physical Layer Testing," September 2012. [Online]. Available: http://grouper.ieee.org/groups/802/3/RTPGE/public/sept12/jones_01_0912.pdf. [Accessed 19 September 2012].

[56] D. Pannell, "Passing Automotive EMC with IEEE Standard PHYs," September 2012. [Online]. Available: http://grouper.ieee.org/groups/802/3/RTPGE/public/sept12/pannell_01_0912.pdf. [Accessed 19 September 2012].

[57] LegalForce Trademarkia, "QUIET-WIRE," 24 July 2013. [Online]. Available: www.trademarkia.com/quietwire-85360886.html. [Accessed 22 December 2013].

[58] L. Frenzel, "Networking Technologies Take Charge at Home and on the Road," 8 November 2012. [Online]. Available: http://electronicdesign.com/communications/networking-technologies-take-charge-home-and-road. [Accessed 22 December 2013].

[59] Computer Business Review, "Media-Independent Interface for 100BASE-T Published," 12 April 1994. [Online]. Available: www.cbronline.com/news/media_independent_interface_specification_for_100base_t_published. [Accessed 23 December 2013].

[60] R. Kraus, "APIX2 – a Whole Flood of Data," *Elektronik*, pp. 2–5, July 2011.

[61] H. Zinner, K. Matheus, S. Buntz, and T. Hogenmüller, "Requirements Update for IEEE 802.3 RTPGE," 18 July 2012. [Online]. Available: http://grouper.ieee.org/groups/802/3/RTPGE/public/july12/zinner_02_0712.pdf. [Accessed 18 July 2012].

[62] T. Hogenmüller and H. Zinner, "Tutorial for Wake-Up Schemes and Requirements for Automotive Communication Networks," 18 July 2012. [Online]. Available: http://grouper.ieee.org/groups/802/3/RTPGE/public/july12/hoganmuller_02a_0712.pdf. [Accessed 25 July 2012].

[63] T. Hogenmüller and H. Zinner, "Tutorial for Lifetime Requirements and Physical Testing of Automotive Electronic Control Units (ECUs)," 18 July 2012. [Online]. Available: http://grouper.ieee.org/groups/802/3/RTPGE/public/july12/hoganmuller_01a_0712.pdf. [Accessed 25 July 2012].

[64] IEEE 802.3, "Approved Minutes, IEEE 802.3 Ethernet Working Group Plenary Grand Hyatt, San Antonio, TX USA," 12–15 November 2012. [Online]. Available: www.ieee802.org/3/minutes/nov12/minutes_1112.pdf. [Accessed 7 October 2013].

[65] P. Wang and M. Tazebay, "1000BASE-T1 PHY Synchronization Method & Start-up Process and Link Failure Case Analysis," 15 July 2014. [Online]. Available: www.ieee802.org/3/bp/public/jul14/wang_3bp_01_0714.pdf. [Accessed 11 November 2016].

[66] J. Graba, "EEE Proposal for 1000BASE-T1," 16 July 2014. [Online]. Available: www.ieee802.org/3/bp/public/jul14/Graba_3bp_01a_0714.pdf. [Accessed 11 November 2016].

[67] Wikipedia, "IEEE 802.3," Wikipedia, 20 October 2016. [Online]. Available: http://en.wikipedia.org/wiki/IEEE_802.3. [Accessed 13 November 2016].

[68] F. Idachaba, D. U. Ike, and O. Hope, "Future Trends in Fiber Optics Communication," 2 July 2014. [Online]. Available: www.iaeng.org/publication/WCE2014/WCE2014_pp438-442.pdf. [Accessed 26 March 2016].

[69] Y. Tsukamoto, "Plastic Optical Fiber (POF) technology for Automotive, Home network systems," May 2014. [Online]. Available: www.ieee802.org/3/GEPOFSG/public/May_2014/Tsukamoto_GEPOF_01_0514.pdf. [Accessed 9 July 2016].

[70] K. Matheus et al., "Comparison of Electrical and Optical Transmission for Gbps Ethernet v.1.0," 13 May 2014. [Online]. Available: www.ieee802.org/3/GEPOFSG/public/May_2014/Pardo_GEPOF_1_0514.pdf. [Accessed 13 May 2014].

[71] C. Pardo, T. Lichtenegger, A. Paris, and H. Hirayama, "Gigabit Ethernet over Plastic Optical Fibre, Call for Interest," 27 March 2014. [Online]. Available: www.ieee802.org/3/GEPOFSG/public/CFI/GigPOF%20CFI%20v_1_0.pdf. [Accessed 25 March 2016].

[72] KDPOF, "Simple Introduction to Gigabit Communications over POF, Revision 1.1," 2014. [Online]. Available: www.kdpof.com/wp-content/uploads/2012/07/Easy-Introduction-v1.1.pdf. [Accessed 26 November 2016].

[73] Thorlabs, "Graded-Index Polymer Optical Fiber (GI-POF)," unknown. [Online]. Available: pofto.com/downloads/ieee/pof.branches.v6.doc. [Accessed 25 March 2016].

[74] B. Grow, C. Pardo, N. Serisawa, and T. Lichtenegger, "Tutorial: Gigabit Ethernet over Plastic Optical Fiber (GEPOF)," 3 November 2014. [Online]. Available: www.ieee802.org/3/GEPOFSG/public/Tutorial/tutorial_GEPOF_1d_1114.pdf. [Accessed 25 March 2016].

[75] M. Tomlinson, "New Automatic Equalizer Employing Modulo Arithmetic," *Electronics Letter*, vols. 5–6, no. 7, pp. 138–139, March 1971.

[76] H. Harashima and H. Miyakawa, "Matched-Transmission Technique for Channels with Intersymbol Interference," *IEEE Transactions on Communications*, vol. 20, no. 4, pp. 774–780, August 1972.

[77] IEEE-SA, "IEEE 802.3bv/D2.0 Amendment: Physical Layer Specifications and Management Parameters for 1000 Mb/s Operation over Plastic Optical Fiber," IEEE-SA, New York, 2016.

[78] Wikipedia, Various different pages [Online]. Available: www.wikipedia.org. [Accessed 2016].

[79] MakeUseOf, "EDGE, 3G, H+, Etc: What Are All These Mobile Networks?," February 2016. [Online]. Available: www.makeuseof.com/tag/edge-3g-h-etc-mobile-networks/. [Accessed 26 May 2016].

[80] Wikipedia, "List of Digital Television Deployments by Country," 12 April 2016. [Online]. Available: https://en.wikipedia.org/wiki/List_of_digital_television_deployments_by_country. [Accessed 25 May 2016].

[81] Wikipedia, "Digital Radio," 12 April 2016. [Online]. Available: https://en.wikipedia.org/wiki/Digital_radio. [Accessed 25 May 2016].

[82] K. Barbehön, "Mitten im Umbruch zur Digitalisierung: Wie die Zukunft der E/E Architektur von IT-Standards beeinflusst wird," in *21st International Congress on Advances in Automotive Electronics*, Ludwigsburg, 2016.

[83] S. Carlson, H. Zinner, K. Matheus, N. Wienckowski, and T. Hogenmüller, "CFI Multi-Gig Automotive Ethernet PHY," 9 November 2016. [Online]. Available: www.ieee802.org/3/ad_hoc/ngrates/public/16_11/20161108_CFI.pdf. [Accessed 12 November 2016].

[84] G. Armbrecht, "Plugs and Cables for Data Transmission up to 12 Gbps," in *High-Speed Datenübertragung in Automobilnetzen der Zukunft*, Erlangen, 2015.

[85] H. Adel, "Methodology for Cable Performance Optimization," in *High-Speed Datenübertragung in Automobilnetzen der Zukunft*, Erlangen, 2015.

[86] M. Kuijk, "Qualification of Coax as Physical Layer for in-Car Networks," in *MOST Forum*, Esslingen, 2014.

[87] T. Kimura and M. Dittmann, "Glass Fiber Solution for Multi-Gigabit per Second," in *MOST Interconnectivity Conference Asia*, Tokyo, 2015.

[88] K. Matheus, "The Future of in-Vehicle Networking Technologies – Survey Results," February 2016. [Online]. Available: http://go.hanser-automotive.de/1299263. [Accessed February 2016].

[89] C. Gauthier, "10 Mb/s SPE PHY Feasibility," 20 September 2016. [Online]. Available: http://standards.ieee.org/events/automotive/2016/d1_panel_01_gauthier_omniphy_10base_t1_automotive_v3.pdf. [Accessed 12 November 2016].

[90] M. Tazebay, "Feasibility Framework for 10SPE Automotive," 20 September 2016. [Online]. Available: http://standards.ieee.org/events/automotive/2016/d1_panel_02_tazebay_iptechday_2016_v2.pdf. [Accessed 12 November 2016].

[91] L. Winkel, M. M. Carthy, D. Brandt, G. Zimmerman, D. Hoglund, K. Matheus, and C. DiMinico, "10Mb/s Single Twisted Pair Ethernet Call for Interest," 26 July 2016. [Online]. Available: www.ieee802.org/3/cfi/0716_1/CFI_01_0716.pdf. [Accessed 12 November 2016].

[92] G. Zimmerman, "IEEE 802.310 Mb/s Single Twisted Pair Ethernet (10SPE) Study Group Closing Report," 10 November 2016. [Online]. Available: www.ieee802.org/3/minutes/nov16/1116_10M_stp_close_report.pdf. [Accessed 12 November 2016].

[93] G. Zimmerman, "Adopted Objectives," 10 November 2016. [Online]. Available: www.ieee802.org/3/10SPE/objectives_10SPE_111016.pdf. [Accessed 12 November 2016].

[94] International Energy Agency, "CO_2 Emissions from Fuel Combustion: Highlights," 2015. [Online]. Available: www.iea.org/publications/freepublications/publication/CO2EmissionsFromFuelCombustionHighlights2015.pdf. [Accessed 2 April 2016].

[95] S. C. Davis, S. W. Diegel, and R. G. Boundy, "Transportation Energy Data Book: Edition 30–2011," June 2011. [Online]. Available: http://info.ornl.gov/sites/publications/files/Pub31202.pdf. [Accessed 2 April 2016].

[96] statista, "Number of Passenger Cars and Commercial Vehicles in Use Worldwide from 2006 to 2014 (in 1,000 Units)," 2016. [Online]. Available: www.statista.com/statistics/281134/number-of-vehicles-in-use-worldwide/. [Accessed 2 April 2016].

[97] IEEE 802.3, "Approved Minutes, IEEE 802.3 Ethernet Working Group Plenary, Dallas, TX, 11 to 14 November, 2013," 14 November 2013. [Online]. Available: www.ieee802.org/3/minutes/nov13/1113_minutes.pdf. [Accessed 25 September 2016].

[98] IEEE-SA, "IEEE 802.3bu-2016 Amendment: Physical Layer and Management Parameters for Single-Pair Power over Data Lines (PoDL)," IEEE-SA, New York, 2016–17.

[99] D. Grossmann and H. Hild, "Smart Charging – a Key to Successful E-Mobility," July 2014. [Online]. Available: https://vector.com/portal/medien/cmc/press/Vector/EV_Smart_Charging_ElektronikAutomotive_201407_PressArticle_EN.pdf. [Accessed 24 September 2016].

[100] F. Nouvel, P. Tanguy, S. Pillement, and H. Pham, "Experiments of In-Vehicle Power Line Communications," 11 April 2011. [Online]. Available: http://cdn.intechopen.com/pdfs/14996/InTech-Experiments_of_in_vehicle_power_line_communications.pdf. [Accessed 24 September 2016].

[101] ISO, "ISO 11898-6:2013: Road vehicles – Controller Area Network (CAN) – Part 5: High-Speed Medium Access Unit with Selective Wake-up Functionality," ISO, Geneva, 2007.
[102] S. Kerner, "Energy Efficient Ethernet Hits Standards Milestone," 17 June 2009. [Online]. Available: www.internetnews.com/skerner/2009/07/energy-efficient-ethernet-hits.html. [Accessed 23 December 2013].
[103] A. Liu, P. Axer, and C. Long, "OPEN Sleep/Wake-up Specification," OPEN Alliance, Irvine, 2016.
[104] R. Sappia, "Enabling Power Efficient and Interoperable Behavior in a Multi-vendor Environment of 100BASE-T1 Components," in *Automotive Ethernet Congress*, Munich, 2016.
[105] C. Kymal and P. Patiyasevi, "Semiconductor Quality Initiatives," April 2006. [Online]. Available: www.qualitydigest.com/april06/articles/05_article.shtml. [Accessed 17 December 2013].
[106] Automotive Electronics Council, "AEC History," unknown. [Online]. Available: www.aecouncil.com/AECHistory.html. [Accessed 17 December 2013].
[107] Automotive Electronics Council, "Annual Reliability Workshop," 2013. [Online]. Available: www.aecouncil.com/AECWorkshop.html. [Accessed 17 December 2013].
[108] Broadcom, "OPEN Alliance BroadR-Reach (OABR) Physical Layer Transceiver Specification for Automotive Applications v3.2," 24 June 2014. [Online]. Available: www.ieee802.org/3/1TPCESG/public/BroadR_Reach_Automotive_Spec_V3.2.pdf. [Accessed 5 November 2016].
[109] C. Stroud, "Linear Feedback Shift Registers (LFSRs)," October 2004. [Online]. Available: www.eng.auburn.edu/~strouce/class/elec6250/LFSRs.pdf. [Accessed 24 October 2016].
[110] F. Ruskey, "Information on Primitive and Irreducible Polynomials," 23 May 2011. [Online]. Available: http://theory.cs.uvic.ca/inf/neck/PolyInfo.html. [Accessed 24 October 2016].
[111] R. Ward and T. Molteno, "Table of Linear Feedback Shift Registers," 26 October 2007. [Online]. Available: http://courses.cse.tamu.edu/csce680/walker/lfsr_table.pdf. [Accessed 5 November 2016].
[112] O. Ziemann, J. Krauser, P. E. Zamzow, and W. Daum, *POF Handbook*, Berlin: Springer, 2008.
[113] VDE Verband der Elektrotechnik Elektronik Informationstechnik e.V., "VDE V 0885–763," VDE, Frankfurt, 2012.
[114] Wikipedia, "Paris Agreement," 2 April 2016. [Online]. Available: https://en.wikipedia.org/wiki/Paris_Agreement. [Accessed 3 April 2016].
[115] UNFCCC Newsroom, "New UN Report Synthesizes National Climate Plans from 146 Countries," 30 October 2015. [Online]. Available: http://newsroom.unfccc.int/unfccc-newsroom/indc-synthesis-report-press-release/. [Accessed 3 April 2016].
[116] Wikipedia, "Energy-Efficient Ethernet," 12 September 2016. [Online]. Available: https://en.wikipedia.org/wiki/Energy-Efficient_Ethernet. [Accessed 25 September 2016].
[117] Cisco, "Cisco Greening Routers," 27 January 2009. [Online]. Available: https://web.archive.org/web/20090618035001/http://greenethernet.com/. [Accessed 25 September 2016].
[118] S. Wang, "Modelling and Design of Planar Integrated Magnetic Components," 31 July 2013. [Online]. Available: https://theses.lib.vt.edu/theses/available/etd-08072003-141703/unrestricted/Thesis_Shen.pdf. [Accessed 24 December 2013].

5 Protocols for Automotive Ethernet

One of the reasons for the automotive industry to adopt Ethernet-based communication as an in-vehicle networking system is the chance for synergies, i.e., the possibility of reusing protocols that have been developed and tested in other industries. Across the various protocol layers for the various applications it therefore needs to be carefully investigated whether to adopt, adapt, or to add protocols. Figure 5.1 gives an example overview of a typical protocol stack. This chapter discusses four areas that require special care: Audio Video Bridging (AVB) and its successor Time-Sensitive Networking (TSN, see Section 5.1), Virtual LANs (VLANs) and switch configuration in the context of security (see Section 5.2), the Internet Protocol (IP; see Section 5.3); and what is needed in terms of command and control (see Section 5.4).

Note that the described solutions make no claim to be complete; it might well be possible to use other protocols with the Automotive Ethernet PHY transceivers. However, the solutions described in this section describe a solution that works and that can be adopted by those wanting to deploy Automotive Ethernet.

5.1 Quality of Service (QoS), Audio Video Bridging (AVB), and Time-Sensitive Networking (TSN)

Ethernet as such, i.e., the PHY and MAC layers as defined by IEEE 802.3 at the time, provide best-effort communication only. Introducing switches improved the determinism of each individual link, since the various connected units no longer needed to contend for the same medium at potentially the same time and in case of collisions had to go into random, i.e., nondeterministic, back-off periods. However, in a switched network, data of different sources with different destinations might still have to be sent over the same link at the same time. It is therefore on Layer 2 in the switch – often also referred to as a (multiport) bridge[1] – that Quality of Service (QoS) requirements can effectively be supported. Today, it is mainly at IEEE 802.1 that the respective protocols and procedures are being standardized.

This book uses the term QoS for requirements and solutions that influence the flow of data such that it can be received at a defined quality [2]. These can vary significantly depending on the use case and focus area (see, e.g., [3] [4]). It is therefore important to start with some background information on the origin of the Audio Video Bridging

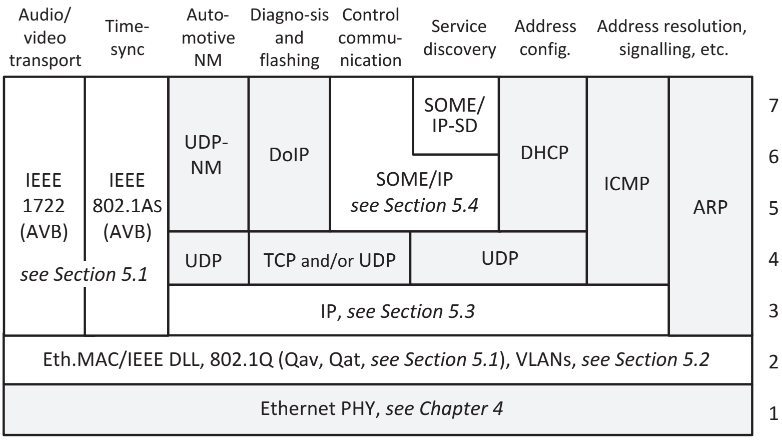

Figure 5.1 Protocol overview for Automotive Ethernet [1].

(AVB) standardization activity (Section 5.1.1) and to highlight the differences between the originally envisioned audio and video use cases and their deployment in automotive (Section 5.1.2). In return, this allows describing how each QoS protocol for audio and video applications can best be used in in-vehicle networking (Section 5.1.3). However, even if audio video entertainment provides the origin for the series of IEEE standards, this is not conclusive. Section 5.1.4 describes efforts around standardizing protocols for more safety critical applications in an Ethernet network. These efforts are called Time-Sensitive Networking (TSN).

5.1.1 How Audio Video Bridging (AVB) Came to Ethernet

In July 2004, the IEEE 802.3 group accepted a Call For Interest (CFI) on “Residential Ethernet” in order to investigate the use of Ethernet for time-sensitive Audio and Video (AV) applications [1]. Apparently, more than a year previously, discussions on the need for more Consumer Electronics (CE) centric Ethernet networking had started simultaneously in different groups of industry players, who then aligned the standardization in the IEEE [5] [6].

At the time, the Internet in combination with audio – and later video – compression formats was drastically and irreversibly changing the consumer behavior in respect to music consumption. Even if the “share it with all for free” Napster platform had only lived from May 1999 to February 2001 [7],[2] it initiated a change: The PC/laptop/mobile device replaced the home hi-fi and CD collection as the center for consumer entertainment. These new devices are able to serve at the same time as storage, rendering device, synthesizer, sound mixer, and media server, and the PCs and laptops – at least up till now [8] – always had an Ethernet interface. The CFI on Residential Ethernet addressed the specific quality requirements of AV transmission in an Ethernet LAN, and thus broadened the market potential of Ethernet into the consumer space.

Next to being widely deployed in PCs and laptops, Ethernet offered [5] plug & play, large data rates, and network management in terms of neighbor discovery, virtual network support, and traffic prioritization. Nevertheless, even with priority, there was/is neither timing guarantee nor a reference time to which the receiver can relate. Buffering data can help to overcome jitter up to a certain point. AV applications like VoIP or IP-TV rely on buffering in order to improve the quality. However, finding the correct buffering size in such a situation is no easy task: Buffers that are too small bear the risk of buffer overflow, dropped packets, and quality degradation, while buffers that are too large are costly and introduce additional latency. So, Ethernet did not support endpoint synchronization, timing support, bounded latency support, and bandwidth allocation [5]. The goal of the IEEE 802.3 effort was thus to develop mechanisms for better supporting AV applications in an Ethernet network by providing the appropriate mechanisms.

Very quickly after the respective study group had been set up by IEEE 802.3, it became apparent that the proposed solutions were better suited for standardization in IEEE 802.1 [9], and by the end of November 2005 the effort was officially moved [10]. In September 2011 (plus a latecomer in August 2013), the following set of standards associated with first-generation Audio Video Bridging (AVB, AVBgen1) were completed (see also Section 5.1.3):

- IEEE 802.1Qav, "Forwarding and Queuing Enhancements for Time-Sensitive Streams" (traffic shaping), 5 January 2010[3]
- IEEE 802.1Qat, "Stream Reservation Protocol," 30 September 2010
- IEEE 802.1AS, "Timing and Synchronization for Time-Sensitive Applications," 30 March 2011
- IEEE 1733, "Protocol for Time-Sensitive Applications in Local Area Networks" (AVB adaptation of RTP), 25 April 2011
- IEEE 1722, "Transport Protocol for Time-Sensitive Applications in a Bridged Local Area Network" (layer 2 transport protocol), 4 March 2016
- IEEE 802.1BA, "Audio Video Bridging (AVB) System" (overall system configuration, profiles), 30 September 2011
- IEEE 1722.1, "Device Discovery, Connection Management, and Control Protocol for 1722™ Based Devices" (control mechanisms and service discovery), 23 August 2013

These standards leave the implementer with a variety of options. In 2008, a number of companies participating in the AVB standardization at IEEE thus started working on the formation of an industry alliance that would market AVB and aid interoperability [11]. The AVnu Alliance that was launched in consequence in August 2009 [12] now offers the respective certification programs. Concerning the proliferation of AVB in the market, AVB enabled silicon started to emerge in 2012 (see, e.g., [13] [14]), which from an automotive perspective was just right. With finishing the AVB standards listed above, the IEEE started directly with standardizing AVGgen2 in order to be able to support more time-critical applications (see also Section 5.1.4). As these do not longer comprise audio and video applications only, the AVB effort was officially renamed Time-Sensitive Networking (TSN)in November 2012 [15].

5.1.2 The Audio Video Bridging (AVB) Use Cases

Before going into detail of specific applications, some general remarks on fundamental differences between the quality requirements of AV consumption (including speech) and those traditional Ethernet was designed for:

- While most data applications require every single packet to arrive intact, the occasional **packet loss** can go unnoticed by the user in the case of AV transmissions. That not all information is needed, is emphasized as such by the fact that many AV compression formats are "lossy," meaning that not all information can be recovered with the decompression. MP3 and MPEG are well-known examples. Even though it is in general more critical to lose a packet of compressed data than a packet of uncompressed data, an additional and occasional loss of a packet containing compressed data is not necessarily perceived as a quality degradation [16].
- The situation reverses in the case of **delays**. In general, delays in the milliseconds range or even of a few seconds not discussed when it concerns a file transfer or building up a website. AV applications, in contrast, have very stringent timing requirements [17]:
  - The **absolute delay** must be small in the case of live streaming data. A musician, e.g., tolerates only a 10 ms delay between initiating a sound and expecting to hear it [18] [19].
  - The data must be **synchronized**. In a home (or concert hall) sound and image might travel different paths with different delays. For quality replay, the delay between sound and picture replay needs to be smaller than ±80 ms ("lipsynch" requirement) [20]. Standard home stereo sound needs synchronization between the different streams of less than ±1 ms [18]. For high-end surround sound the synchronization requirement reduces to less than ±10 μs [20], or ±1 μs in the professional environment [9].
  - No matter where the AV content is stored or replayed, there should be **no noticeable jitter**. Sudden interruptions or delay variations in AV streams can occur in case competing traffic is taking up too much data rate. Buffers can only partially compensate for this. For some applications the absolute delay is simply limited and in general the larger the buffer the higher the costs.

5.1.2.1 In Homes/Consumer Devices

Since the introduction of audio and video (AV) compression formats, AV streams are turned into strings of data packets that can be stored on and replayed from various consumer devices like PCs, laptops, tablets, phones, memory-sticks, portable music players, etc. While the traditional home entertainment consisted of units with a clear media-to-function relation (record player, tape recorder, CD-player, amplifier, etc.) and one-way, analog communication between them, the transformation of entertainment data into packets allows for/requires bidirectional networking between units, which Ethernet inherently supports. The general observations made above apply to all AV applications.

Additional requirements, e.g., on timing in consumer devices come from gaming applications, which require a response time of less than 50 ms for human activity and less than ±80 ms difference between video animation and audio. Other home related AV applications with again different requirements are home surveillance and health care [5].

In contrast to the professional audio and automotive use cases, requirements that prevail in consumer applications are (a) the needed support of ad hoc/plug & play capability (no IT administrator) and (b) the necessity of lowest cost [5]. In consequence, aspects like discovery (which is addressed on higher layers by UPnP[4] and DLNA[5]), self-configuration, and a high level of compatibility and interoperability are very important [11]. Furthermore, in a home, media might be shared over a variety of networks that include IEEE 802.11 WLAN/WiFi and Coordinated Shared Networks[6] (CSN) [21], as well as Ethernet.

5.1.2.2 In Professional Audio

Typical application areas for networked professional audio equipment are concerts/live shows, recording studios, conference centers, theme parks, houses of worship, art installations, or any other place where live sound is used professionally [17] [22]. This emphasizes one of the fundamental differences to the consumer use case: In professional audio, good quality perception is a core purpose. A network deployed for professional audio has to be absolutely reliable, with no audio defects, video dropouts, or other artifacts [17]. Furthermore, the timing requirements are very stringent: As has been said, for musicians the delay, e.g., between the microphone and the earphone of an artist needs to be smaller than 10 ms [18] [19]. Allowing 8 ms for processing means that the network delay cannot exceed 2 ms [23]. Professional audio also has very stringent requirements on speaker synchronization, which needs to be within a few microseconds [9].

As with all industrial products the use of a new technology/concept needs to provide for direct or indirect cost savings (or for new functionalities that are expected to result in monetary advantages, see also Figure 3.9. The starting point of professional audio networks is a setup that comprises a huge amount of high-quality, single-purpose, unidirectional, analog, or even digital audio cabling using different technologies. Furthermore, the same extensive wiring is used for the respective video infrastructure, and yet another lot of cabling for control (of amplifiers and loudspeakers), which might use an Ethernet infrastructure [9] [17] [22] [24] [25]. This is not only expensive in respect to the wiring, but also difficult to maintain, and invites the development of proprietary solutions on higher layers, which seem to have prevailed for a long time [5]. Being able to handle such a setup requires very specialized know-how [17].

Thus, the attraction to be able to use a single network, i.e., the Ethernet infrastructure, for all data that need to be networked in the professional AV applications, is obviously large. In pre-AVB times, this was too cumbersome [9] [17]. So when AVB activities started, it is only natural that these were supported by professional audio companies from the start [18], in the IEEE [5] as well as when establishing the AVnu Alliance [11].

An important difference to the residential/consumer and automotive uses of AVB is that the extent of the professional audio network can be significantly larger, in meters as well as in number of nodes. However, in contrast to the consumer use case, the professional AVB network can be expected to be professionally set up and controlled.

5.1.2.3 In Cars

Ever since Ethernet started being discussed for automotive use, the Quality of Service (QoS) capabilities of Ethernet and the potential of the AVB solutions have been investigated (see also Section 3.2). With Ethernet coming from the IT and CE industries, Ethernet was first considered for "similar" in-vehicle infotainment applications. So, while today, Ethernet is naturally being discussed also for in-vehicle control applications (see Section 5.1.4) the focus at the beginning was on enabling AV applications and by the time the AVnu Alliance was set up in 2009, automotive applications were identified as one of the target areas for AVB [26].[7]

In-vehicle AV consumption was not one of the original use cases addressed with the standardization of AVB at IEEE. Nevertheless, Infotainment is an important quality element for vehicle users; after all, the stringent automotive Electromagnetic Compatibility (EMC) requirements (see also Section 4.1) have also been installed to ensure unblemished audio consumption while driving. Nevertheless, in the hierarchy of applications inside vehicles, infotainment is always secondary in relation to driving and safety. This is the most important difference to the consumer and professional audio use cases discussed before and its consequences impact the in-vehicle use of AVB (see the following sections). Furthermore, automotive has an additional timing constraint: The AV system needs to be fully operational within two seconds of power on [27]. Neither in the consumer domain, nor in professional audio does such a (stringent) start-up requirement exist. As in the professional audio domain, the in-vehicle AV network is professionally set up beforehand, even if various car models exist and the exact layout additionally depends on the options the customer selects.

Naturally, also the automotive audio use case itself differs from the ones that can be expected in homes or even the professional environment. The high-quality expectations from car customers and the complexity of handling the various different audio sources in vehicles had once even led to the development of a new in-vehicle networking technology (MOST, see Section 2.2.4.1). An example of audio use cases and their hierarchy inside the vehicle is presented in Table 5.1. As can be seen, a significant amount of the complexity is not handled at the network interface but is organized on higher layers. These can be based on the same principles that MOST handles the functionalities or use a GENIVI-based implementation (see Section 3.5.3), which in return also supports MOST. From an automotive perspective, it is important to keep the separation between application specific requirements and QoS functions the network can provide based on AVB. The separation of the ISO/OSI layering model should be maintained.

Furthermore, costs and resources distinguish the use cases. The AVB functionalities require hardware capabilities in the Ethernet semiconductors and processing power from a separated microcontroller (µC) or from the switch. In professional audio it can be assumed that all processing resources needed will be provided. After all, audio function and quality is their prime concern. In the CE industry costs, in principle, need to be low, though the resources available and customer expectations are likely to vary significantly,

Table 5.1 Example audio hierarchy in an automotive audio network

Layer	Functional block	Features and functions
High	Human–Machine Interface (HMI)	Customer interface for volume control, source changes, additional control interfaces (e.g., changes of volumes for audio interrupts sources like jingles and alarms).
Mid	Audio management system	Fixed system behavior: controls the mixing stages in the audio sink by special control commands. An example is the audio output in case a navigation audio guidance message or park control beep occurs at the same time as the driver is listening to the radio or making a phone call via the in-built hands-free system. The solutions here are generally car manufacturer specific.
Low	Audio network interface	Network resource management: responsible for the availability of the requested bandwidth. The source needs to know when to allocate bandwidth and the sink knows when and how to connect to the source data.

depending on the monetary value spent on the CE device. However, a laptop or even a tablet will have much greater resources available than a typical ECU inside a car that is optimized for cost, space, and processing power.

While costs are important, the question of where and how to implement the AVB functions in an automotive network has more facets. If the AVB functions are embedded on a microcontroller that is integrated into the switch, which supplier will provide the software for it, the Tier 1 or the semiconductor supplier? If the AVB software comes from the semiconductor supplier, but the Tier 1 is responsible for the function of the ECU, who, if there is a malfunction, can diagnose it? Who is responsible? In case resources of the ECU's main purpose μC are used, how can it be ensured that the network functionality is never impaired by other ECU functions, especially during start-up or reboots? As a first approach, [28] proposed to use only an absolute minimum of AVB features for the automotive networks and with this initiated an important discussion on automotive AVB. But the result was nonstandard compliant, and was therefore developed further [27]. As a guideline, it is helpful to store as much (initial) configuration data as possible in some form of digital memory, in order to achieve independence between the ECU and networking functions, especially during start-up. For details, see Section 5.1.3.

5.1.2.4 Direct Comparison of Use Areas

The description in the previous sections showed that the three use cases have very different requirements. What they have in common is that all would like to realize high-quality AV streaming in a (mainly) Ethernet-based network. Additionally, the use and the network are restricted to a certain purpose, size, and physical location, even if a concert hall network can have significantly larger dimensions than a LAN inside a family home or an "Automotive Area Network" (AAN) inside a car. All three use cases can live with

Table 5.2 Comparison of the requirements and properties of the different AVB application areas

Criteria	Home/consumer devices	Professional audio	Car AV
AV application scenarios	Multiple source/sink AV replay in the home, home surveillance [5]	Recording studios, concerts/live shows, conference centers, theme parks, houses of worship, art installations [17] [22]	Simultaneous audio streams of different priority, synchronous replay of AV, camera data for driver assistance
Importance of AV quality	Expectations are likely to correlate with the price paid for the equipment	**The core purpose**	Entertainment and comfort are important under normal circumstances but **driving (safety) is requirement No. 1**
Variability and planability of network setup	Ad hoc, **plug and play**, no IT admin [5], requires self-configuration, service discovery, etc. [21]	Setup can change but will be carefully planned from event to event	Known number of predefinable variations per car model (limited plug and play from passengers)
Network technologies used	Ethernet, WiFi/WLAN, Coordinated Shared Networks (CSNs) [21]	Ethernet	Ethernet
Service rejection (not enough data rate)	Acceptable	Not acceptable	Not acceptable
Synchroniza-tion accuracy required	Stereo synch ~ 1 ms/10 μs [18] [20]	~**1 μs** [9]	Like for consumer devices
Maximum network delays	50 ms [19]	2 ms application delay [23]	80 ms lipsynch [19]
Start-up requirements	None	None	**AV system fully operational within 2 s** [27]
Link length	<200 m [5]	Can be long, but <1 km expected	≤15 m [29], 3.5 m average [30]
Available processing resources	Depends, larger on computers, smaller on mobile devices	As large as needed	Generally shared with other functions, rather small
Costs	Very low costs [5]	Function before costs, savings in harness	Needs to be competitive (see Section 3.4.2)

the maximum delay guaranteed for 100 Mbps Ethernet AVB traffic of 2 ms over 7 hops on ISO/OSI layer 2.[8]

Table 5.2 directly compares the main properties and requirements of the AVB use cases. The main difference is the low-cost, multivendor plug & play requirement of the consumer use case, in contrast to the very high-quality requirements of the professional audio use case, or the secondary nature of AV in automotive and its stringent start-up requirement.

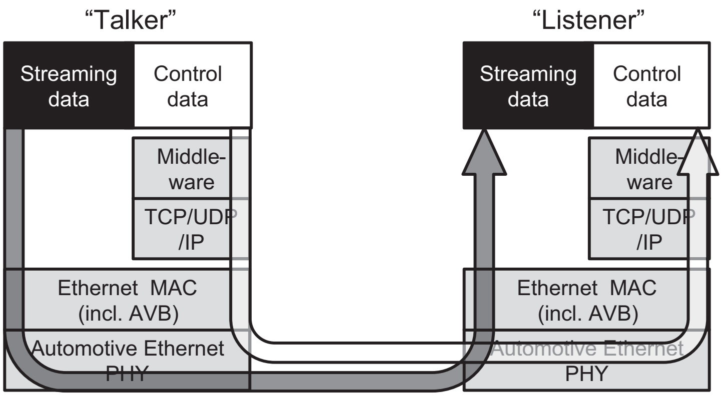

Figure 5.2 IEEE 1722 streaming data and application control data in automotive.

5.1.3 The AVB Protocols and Their Use in Automotive

The AVB specifications facilitate QoS guarantees for streaming data within an "AVB cloud," i.e., a group of networked devices all supporting the core AVB functions either in the role of forwarding switches (which in the IEEE context are referred to as bridges, see introduction to Section 5.1) and/or as end nodes. The basic QoS requirements are that the streams can be rendered in synch with each other, that network delays are not noticeable in the application, and that the network resources are available for as long as the application needs them [9]. AVB distinguishes between "Talkers" that are the source of the streaming data, "Listeners" that are the consumers of those streams, and the AVB capable switches in between. The implementation of AVB requires that the underlying Ethernet network runs at least at 100 Mbps full duplex, that the Ethernet payload does not exceed the maximum size of 1500 bytes, and that the flow control/pause frames (see Section 1.2.1) are disabled. The following subsections describe the AVB mechanisms in more detail.

5.1.3.1 IEEE 1722: Transport

The IEEE 1722 [31] specification describes the transport for AV data. It leverages concepts from the IEC 61883 standards on digital interfaces of consumer AV equipment and thus FireWire/IEEE 1394 [24]. The key property of IEEE 1722 is that it identifies Ethernet packets carrying AV content on layer 2 and not on higher layers. This allows bypassing higher layer protocols (see Figure 5.2), thus reducing the processing time needed and making the latency more predictable.

In principle, IEEE 1722 allows transporting two types of content: streaming data and data for controlling IEEE 1722. Figure 5.3 depicts the respective packet structures: an Ethernet frame/packet carrying an IEEE 1722 packet and its content, which has its own header for the packet and potentially even a header attached to every AV content unit. The Ethernet packet has to include the otherwise optional IEEE 802.1Q header. The priority information (as defined in IEEE 802.1Q and used in IEEE 802.1Qat/SRP) encapsulated within is essential for the functioning of the AVB QoS concept. Using

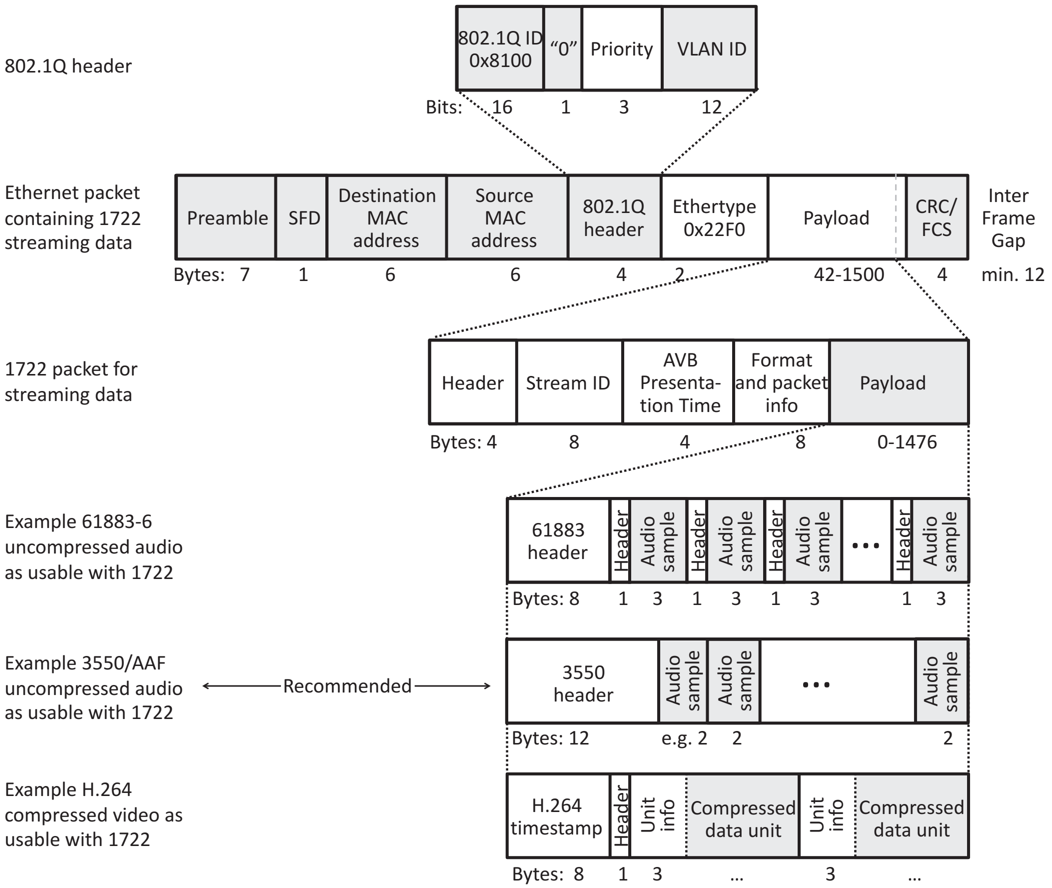

Figure 5.3 IEEE 1722 packet format with example 1722 payloads.

VLANs (see also Section 5.2.2.1) is in principle orthogonal to AVB and independent. However, to be able to receive streams Listeners must be members of the right Talker's VLAN [31]. The IEEE 1722 Ethertype is 0x22F0. If the IEEE 1722 packet is shorter than the required 42 bytes minimum length, the Ethernet MAC will pad the packet automatically, like for any other protocol transmitted via Ethernet.

An IEEE 1722 streaming packet consists of a header, the stream ID, the "Presentation Time," payload information and the payload itself (see Figure 5.3). The header defines what type/format of AV data to expect. It also includes the sequence number in order to allow Listeners to identify missing packets. The stream ID unambiguously defines a specific data stream and is derived from the Talker's MAC address (see Section 5.1.3.3). The field for payload information is directly related to the format of the data inside the payload.

A very important part of AVBTP 1722 is the "Presentation Time." It defines the time a received packet should be presented to the Listener application, i.e., when it should leave layer 2 at the receiver. The Talker sets the Presentation Time depending

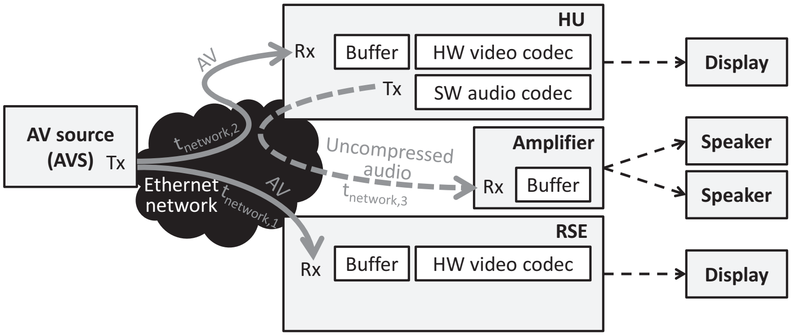

Figure 5.4 Different audio and video paths in an Automotive Area Network (AAN) that are a challenge for achieving lipsynch.

on the time the packet left the application buffer inside the Talker and the expected worst-case duration the packet needs in the cloud ("Max Transit Time"). The default value for the Max Transit Time in the case of the highest priority streaming "Class A" traffic is 2 ms; in the case of the next lower priority, "Class B" traffic, it is 50 ms. The standard allows this value to be different/negotiated, but it does not define how this should be done. Standard plug & play equipment can thus be expected to use the standardized default value(s). The Presentation Time is represented in nanoseconds (ns) as the remainder when dividing the absolute time by $2^{32} - 1$ ns. The concept of the Presentation Time is a good example of the close interrelationship between the AVB standards, as the Presentation Time can only work in a previously synchronized network (see Section 5.1.3.2 for IEEE 802.1AS). Once the synchronization has been established the Presentation Time can be used as feedback and correction for the synchronization. The Max Transit Time is also one of the values that determine the required buffer size of the Listeners (see also Section 5.1.3.3).

The IEEE 1722 provides an important mechanism also for QoS in Automotive Ethernet. In terms of its use, the following considerations are important:

1. **Supported data formats:** The IEEE 1722–2011 specification covered mainly FireWire/ISO 61883 headers[9] but not, e.g., the formats discussed in the automotive industry for the camera use cases: MJPEG and H.264 (see also ISO 17215). This was changed with the IEEE 1722–2016 release [31]. Should yet more formats be needed, the payloads defined for the Real-time Transport Protocol (RTP) in IETF RFC 3550 [32] can be used with IEEE 1722 without modification.
2. **Use of the Presentation Time:** Figure 5.4 shows an example in-vehicle network consisting of an Audio Video Source (AVS), a Head Unit (HU), a Rear Seat Entertainment (RSE), and an Amplifier. The goal is to have a lip synchronous replay of the content on two different displays and two speakers. The Presentation Time as defined in IEEE 1722 defines the time the data shall be presented to the system beyond layer 2. In the example this means the time the data are passed on to the AV codecs, and not, as would be desirable, the time of playing the data on the displays and speakers. The example scenario is even more critical, as after decoding the uncompressed

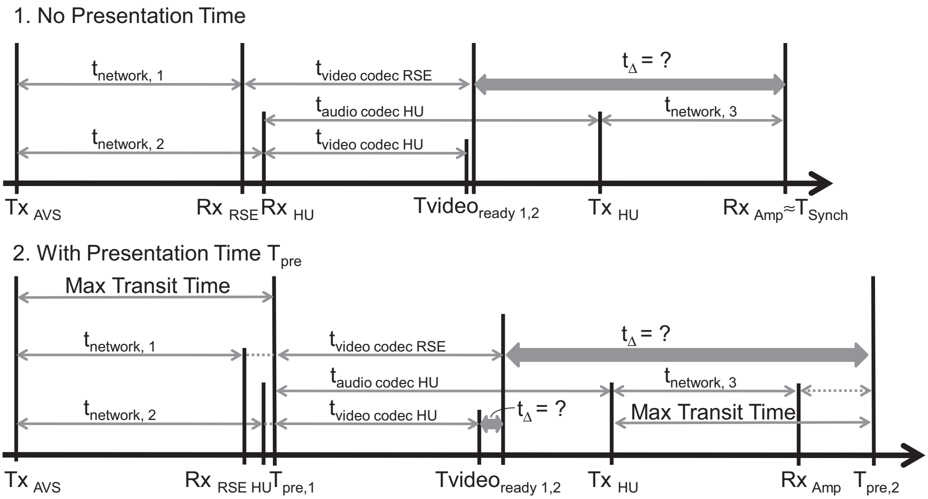

Figure 5.5 Timing behavior with and without Presentation Time T_{pre}, assuming T_{pre} is derived from the maximum delay in the network Max Transit Time.

data is reinserted into the network. This transmission is completely independent from the first and not considered in the original Presentation Time provided by the AVS. The Presentation Time as originally defined is thus not sufficient for ensuring synchronous replay in this scenario. But, even if only the HU and RSE video replay was considered, the Presentation Time would only help, if the processing delay caused by the video decoding in the two units differed only marginally.

Figure 5.5 depicts the consequences for the timing behavior. In the upper half of Figure 5.5, the Presentation Time is not used and the codecs start processing the data the moment they receive it (Rx_{RSE}, Rx_{HU}). In the example, the audio codec is realized in software and slower (without this being decisive). In the end neither the two image streams are ready at the same time ($Tv_{ready,1}$, $Tv_{ready,2}$), nor is the audio, which arrives significantly later at the speakers (Rx_{Amp}). The lower half of the picture shows the same scenario under the assumption that the maximum delay possible, the Max Transit Time, is used to derive the Presentation Time ($T_{pre,1}$, $T_{pre,2}$). As can be seen, this does not help to improve the synchronization between any of the output files.

Instead of a Presentation Time that defines when to present the data to the application beyond layer 2, it would thus be desirable to have a time available that defines when to present the AV information to the customer [27].One approach would be to set the Max Transit Time to a value other than the default values of the standard. The standard, in principle, allows for this. However, it needs to be assured that all used units support the use of a different value.[10]

3 **Dynamic versus static use:** IEEE 1722 expects either locally administered unicast addresses or the use of multicast addresses, which can be statically or dynamically allocated. In order to support the dynamic allocation of the multicast addresses the

IEEE 1722 specification defines a MAC Address Acquisition Protocol (MAAP) [31]. Within a specifically reserved range of multicast addresses the MAAP can dynamically establish which addresses can be used for a new stream, while defending the address from other uses once it has been selected. As has been described with the use cases in Section 5.1.2, the automotive scenario is not particularly dynamic. Once a car model has been designed, the number of different Automotive Ethernet network topologies for this car is limited. At the same time, start-up is critical and therefore all dynamic negotiations disadvantageous. A static preconfiguration is thus also preferred for the IEEE 1722 address allocation.

5.1.3.2 IEEE 802.1AS: Precision Time Protocol (PTP)-Based Synchronization

The main purpose of IEEE 802.1AS [33] is to synchronize all nodes in an AVB cloud to a common reference time. The standard mandates a precision of ±500 ns for two end nodes that have fewer than 7 AVB nodes in between, which means that direct neighbors have to synchronize with nanosecond precision [18] [24]. IEEE 802.1AS – also referred to as the "generalized PTP" (gPTP) – is a simplified extension of the IEEE 1588 specification [34], which had been started at the end of the 1990s and was first completed in 2002 [35] and updated in 2008 (PTPv2). The main difference between IEEE 1588 and IEEE 802.1AS is that gPTP assumes that all communication between time-aware systems is done using IEEE 802 MAC PDUs and addressing only, while IEEE 1588 supports various layer 2 and layer 3–4 communication methods.

Like in most time synchronization approaches, one node in an IEEE 802.1AS network functions as "Grandmaster," to whose clock all other clocks synchronize. Which unit needs to be the Grandmaster is not standardized. Ideally, the Grandmaster is the node with the best suited clock in the cloud. The standard consequently addresses the following two topics: (a) how to select the Grandmaster and (b) how to correctly synchronize to its time throughout the AVB network.

The Grandmaster can be pre- or autoselected with help of the Best Master Clock Algorithm (BMCA). The BMCA is a distributed algorithm: every Grandmaster capable node receiving a respective "announce" message compares the information of the current best Grandmaster with its own clock related quality values. If the eight differently rated values of its own clock yield a better result than that of the current Grandmaster announced in the message, the unit having done the comparison announces itself as the new best Grandmaster. The process is repeated until the truly best Grandmaster in the network has been found. The "announce" messages are sent cyclically and the Grandmaster can change anytime the Grandmaster selection data changes, like the actual Grandmaster leaving the network, a unit having access to a better clock,[11] or a new Grandmaster suitable unit joining.

As a side effect, the BMCA also determines the "clock spanning tree," i.e., the paths on which synchronization related messages pass through the network. This is done by labeling the all ports in the AVB network as follows:

- "Slave port" (the port on which the last message from the Grandmaster was received)
- "Master port" (ports on which the message was passed on)

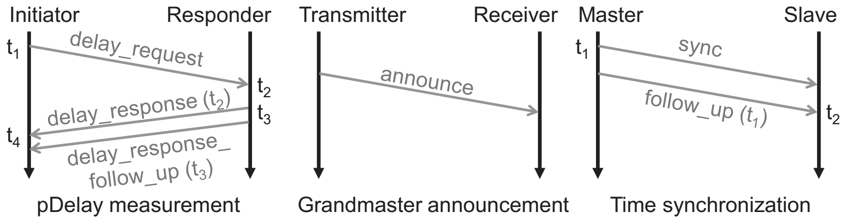

Figure 5.6 Flow charts of major IEEE 802.1AS functions.

- "Disabled port" (ports that connect to non-1AS capable nodes)
- "Passive port" (ports that lead to redundant paths in the AVB cloud).

Normally, an ECU without a switch has one port only. Unless this ECU provides the Grandmaster – then the one port is Master – such nodes have one Slave port only.

In order to achieve synchronization, every unit needs to know the delays caused by the propagation of messages in the network. IEEE 802.1AS thus defines so-called "pDelay" measurements, in which every node in the AVB cloud learns the propagation delays between itself and its direct AVB neighbors to which packets might be sent. Additionally, the pDelay measurement is also suited to determine whether a direct neighbor is actually 1AS capable. An important tool for the pDelay determination is time-stamping: The IEEE 802.1AS Ethertype (0x88F7) triggers sampling of the local clock at the ingress and egress respectively of the PTP message from the MAC [24]. To achieve the aspired nanosecond precision in the time-stamping, it is necessary to implement the time-stamping in hardware instead of software [36]. The pDelay measurements are cyclically repeated.

Last, but not least, the time to synchronize to needs to be made known in the network. This is done in two steps with "sync" and "follow_up" messages along the clock spanning tree in the network. Figure 5.6 gives an overview of the mechanisms.

IEEE 802.1AS messages use a specific multicast MAC address (01-80-C2-00-00-0E) [33]. This address enables units to exchange information between direct neighbors (only). In consequence, it is not foreseen that such IEEE 802.1AS Ethernet packets include the optional IEEE 802.1Q header, as a VLAN information would potentially collide with the purpose of the multicast address used.

For automotive use, it suggests itself to preselect the IEEE 802.1AS Grandmaster (and the clock spanning tree) and to choose as the Grandmaster an ECU every car is equipped with. It may sometimes be the case that an optional ECU would actually provide the better clock, e.g., if the option is equipped with GPS capabilities (mind though that GPS might be problematic in garages and that the clocks used, e.g., in a FlexRay node are also of high quality and well suited). Nevertheless, dynamic selection of the Grandmaster would not only unnecessarily strain the start-up time, it would also lead to more effort in the qualification and testing of the network. More variants (of, e.g., who the Grandmaster is) create more possibilities for malfunctions and robustness is essential in automotive.

The layout of in-vehicle networks is predesigned, known, and in principle does not change during use. Some links and ECUs might be disabled when not in use (see partial networking in Section 6.3.3), like a surround view system that is initialized with the reverse gear and switched off when the velocity of the car has exceeded a certain limit. Nevertheless, the link lengths and the location in the network do not change from one time the car is used to another. Thus, also the pDelay values will not change (much) every time the car is started. In order to speed up the start-up further it is thus proposed to learn and store the last pDelay values.

5.1.3.3 IEEE 802.1Qat: Stream Reservation

The IEEE 802.1Qat Stream Reservation Protocol (SRP) [37] allows allocating bandwidth for individual application and traffic streams within the AVB cloud. The main idea is that Talkers announce the availability of streaming data to all units in the AVB cloud. If Listeners would like to receive the stream, they also announce this. In consequence, all switches the data has to pass through in order to get from the Talker to the Listeners, evaluate the availability of the needed bandwidth. If available, the specific streaming bandwidth is guaranteed. If not available, the reservation request is declined. By default a maximum of 75% of the available bandwidth can be reserved, though system designers can, with care, increase or decrease this number depending on the actual requirements. If a setup allows, e.g., up to 50% of bandwidth to be reserved for Class A traffic, but actually only 30% have been reserved, Class B traffic can reserve the remaining 20% should the available bandwidth in Class B not be sufficient.

Also IEEE 802.1Qat uses special Ethertypes: 0x22EA is used for the actual reservation with the “Multiple Stream Reservation Protocol” (MSRP); 0x88F5 (MVRP) and 0x88F6 (MMRP) identify control packets for necessary information and registration associated with it.[12]

Without limitation, important information needed for the stream reservation is a unique stream ID, which is generated from the Talker’s MAC address and a 16 bit number the Talker assigns to the stream, and quality data about the stream itself. This includes the traffic class, the frame rate, and the length of every packet sent. Table 5.3 gives examples for the bandwidth needed for uncompressed stereo audio data for the defined traffic Classes A and B and a new traffic Class C[13]. This new traffic class has been generated in order to meet today’s DSPs and DMAs typical process block rate of 32 or 64 audio samples at either 44.1 kHz or 48 kHz [38]. As can be seen, transmitting audio samples with the originally defined IEC 61883–6 packet format consumes significantly more bandwidth than using the newer AVTP Audio Format (AAF). The AVnu automotive profile thus recommends the use of the AAF and does not support IEC 61883–6 [38]. Table 5.3 also shows that the higher the frequency of packets the larger the bandwidth consumed.

The use of SRP in automotive poses three main challenges:

1 Table 5.3 shows that the average number of bytes streamed in a Class A packet is very small (for the assumed simple stereo audio use case) and packets are sent at short intervals. Having significantly more overhead than payload is not only a waste

Table 5.3 Examples of the required bandwidth for uncompressed, stereo audio streams in case of different traffic classes and sample rates in an IEC61883–6 and an AAF packet format

Class	Frequency of packets (kHz)	Time between packets (ms)	Number of audio samples per packet (stereo)		Date rate for IEC 61883–6 (Mbps)		Date rate for AAF (Mbps)	
			44.1 kHz	48 kHz	44.1 kHz	48 kHz	44.1 kHz	48 kHz
A	8	0.125	12	12	7.04	7.04	5.76	5.76
B	4	0.25	23	24	4.93	5.06	3.58	3.65
C (64 samples)	0.75/0.689	1.333/1.451	128	128	3.16	3.44	1.78	1.93

Note: Traffic Class C is not part of IEEE 802.1Qat [37]. The class with 64 samples at 44.1 kHz or 48 kHz has been published in [38]. The calculation includes 16-bit sampling, 30 bytes overhead for the Ethernet packet, 24 bytes overhead for the 1722 packet, 8 bytes + 1 byte overhead per sample for IEC 61883 or 12 bytes, and none per sample overhead for AAF.

of bandwidth, it also results in an unnecessarily high processing load in all involved devices. Seen from this perspective, it is more advantageous to use longer packets; ideally they use the complete maximum MAC frame size [27]. To somewhat improve the ratio, it is thus advantageous to introduce new traffic classes, like the one identified with “C” in Table 5.3.

2 A denial of an IEEE 802.1Qat reservation request inside a car is not acceptable. It would be disastrous if, e.g., a driver assist function using camera data failed, because a rear seat passenger was watching a High-Definition (HD) video. Even rejecting an audio stream from a passenger’s mobile device is critical, as it would likely be perceived as a malfunction of the car [27]. It is therefore essential that the AVB network and the expected transmission rates, including the ones from consumer devices, are carefully planned upfront and that these considerations are reflected in the network design (see also Chapter 6).

3 Last, but not least, like the dynamic selection of the Grandmaster, a dynamic reservation of streams at system start-up potentially takes too long. Inside cars, the applications using the Ethernet network as well as their data rate requirements are known; this includes the multimedia applications passengers might bring in. After all, every car only seats a certain number of passengers who each can use only a limited number of devices. It is thus in principle possible to envision a static reservation of streams. However, the information on the reserved streams needs to be provided to all nodes involved. One possible solution is to prestore data that can be accessed with every start-up. There are two ways to generate the prestored data. One is to run the SRP protocol once as part of the network setup during the manufacturing process and to store the outcome. The other is to separately design the data in the development process and store different tables for every car/option combination in every AVB ECU.

5.1.3.4 IEEE 802.1Qav: Forwarding, Queuing, and Traffic Shaping

The idea of IEEE 802.1Qav [37] [39] is to improve the actual quality of all AV transmissions in the AVB network by ensuring that the packets of each individual stream are evenly distributed over time. Even if, e.g., a stream cannot use more than the assigned 10% of the available bandwidth, it makes a significant difference for (the delays of) the rest of the streams (and the network as such), whether this stream uses all bandwidth for 1 minute and then sends nothing for 9 minutes or whether it sends something for 0.1 ms and then nothing for 0.9 ms.

To achieve this is relatively straightforward at the source of streams, the Talkers. The traffic classes assigned with the streams determine the frequency of the packets sent; every 125 μs with Class A, every 250 μs with Class B, and every 1 ms with Class C. It just needs to be assured that the Talkers do indeed send the packets at this rate. For live streams like camera or microphone data or for CD audio the data is generated continuously at the application data rate anyway, so all that needs to be done is to package the data at the assigned packet frequency. In other cases of prestored streaming data, this can be quite different. The transmitting node will likely send as much data as possible at once, with the idea that the receiver will buffer the data until it is played. This unnecessarily strains the bandwidth available in the network at the switches and causes the risk of dropped packets owing to buffer overflow. Larger buffers can mitigate this risk, but increase the costs of the receiver [9] and the latency. So, in the case of prestored streaming data, pacing the output of the Talker can make an essential difference to the AV quality in the network.

In the switches, the situation is more complex. A switch potentially needs to handle AV prioritized streams of multiple Talkers and traffic that passes through from connected non-AVB units. The issue with the latter is the following: Within an AVB cloud, the Stream Reservation (SR) traffic classes are matched to some of the eight quality values provided with IEEE 802.1p; by default Class A traffic has priority 3 and Class B traffic has priority 2. Traffic that passes through the AVB cloud from connected non-AVB units might use the same priority values. These priority values then need to be changed when the traffic enters and restored when it exits the AVB cloud.

A switch only evaluates the Ethernet header to decide through which output port(s) the data is sent and into which priority queue of the output port(s) the data need to go. All data with the same priority go into the same queue, which generally works on a First In First Out (FIFO) basis, independent from the source of the data. SR traffic has priority over non-SR traffic. Only the traffic of the SR Class queues can be paced/shaped.

The functioning of the AVB credit-based shaper is visualized in Figure 5.7. If SR traffic enters the priority queue of a port currently busy it collects credit at an "idle" rate, which is equal to the overall bandwidth currently reserved for the respective SR traffic class. As soon as the port is no longer occupied and the credit is equal to or larger than 0, the priority packet is transmitted. During transmission the credit reduces at a "send" rate, which equals the transmission rate of the link minus the idle rate. If at the end of the transmission the credit is still equal to or larger than 0 more packets from the

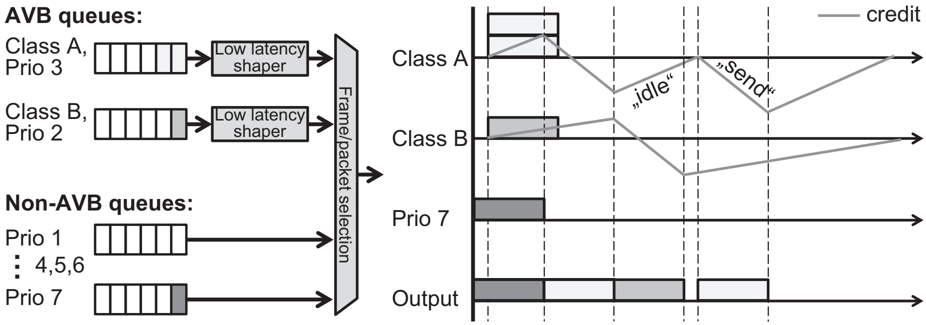

Figure 5.7 Principle of credit-based shaping and queuing with IEEE 802.1Qav.

same queue can be transmitted. If at the end of the transmission the credit is still equal to or larger than 0 and the priority queue is empty, the credit is set to 0. When the credit for a queue is negative, no packet from that queue can be transmitted until the credit has again increased at the idle rate back to 0 or higher.

The main concern for automotive with IEEE 802.1Qav is that the immediate reception of safety critical control data is more important than the AV quality. To simply use the highest priority SR Class A for control data is not a solution as the traffic shaping might actually delay the transmission of a safety critical message [40] [41]. The use of the highest priority non-AVB traffic class for critical control data would guarantee an average throughput of 20% or more (depending on how much has been reserved for the AVB queues). However, AVB-queue packets within their reserved bandwidth always pass first, and the critical control data might be delayed. There is no simple solution for this. Thus, a significant amount of effort of the actual TSN standardization addresses the requirements for safety critical control traffic (see Section 5.1.4).

Apart from the principal concern addressed above, the advantages of shaping as such have been discussed at length for automotive use cases. It is not the issue to pace the Talker output; this is a low-effort implementation. The concern is in the switches, as not every shaping algorithm is suitable for the use cases that require support. If multiple streams pass a switch, a different algorithm might be optimum for each. The solutions thus require careful design.

Figure 5.8 summarizes the AVB elements proposed in Sections 5.1.3.1 to 5.1.3.4 for use in Automotive Ethernet.

5.1.3.5 Other First-Generation AVB Protocols

The following specifications are part of AVB (gen1) but at the time of writing were of minor relevance for automotive implementations:

- The Real-time Transport Protocol (RTP) describes how to transport audio and video (AV) streams over layer 3/IP-based networks. The standard **IEEE 1733–2011** [42] describes how to map the RTP time with the IEEE 802.1AS Presentation Time [9]

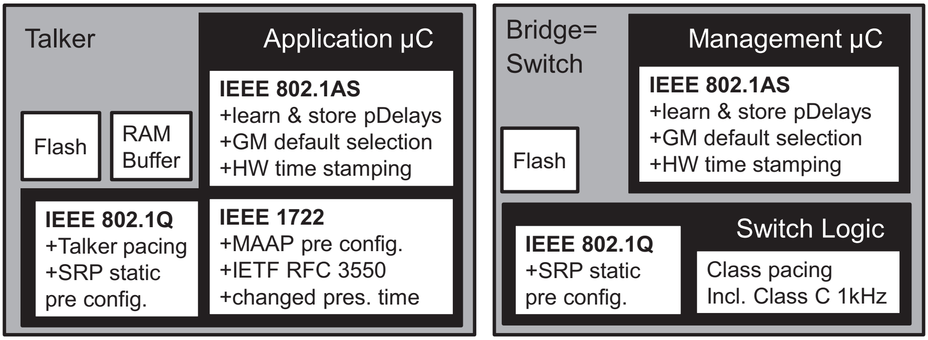

Figure 5.8 Proposed elements of an automotive AVB implementation for automotive AV ECU(s) either in the function of a Talker (Listener) or within the switch semiconductor.

and thus how to let layer 3 AV data profit from layer 2 AVB mechanisms [43]. This increases the flexibility on the technologies used within an AVB network.

- As the previous sections indicated, AVB supports various use cases and ways on how to set up the AVB network. **IEEE 802.1BA-2011** [44] defines profiles and default configurations in order to support easy handling especially where expert network knowledge cannot be expected.
- **IEEE 1722.1–2013** [45] defines typical middleware functionalities for discovering and handling devices and services for IEEE 1722-based systems. From an automotive perspective, the tasks that middleware has to fulfill are more complex than and somewhat different from what is expected when connecting AV multimedia devices. IEEE 1722.1 is therefore of less interest. Table 5.1 showed the hierarchy and different levels of coordination needed when implementing AV applications in cars. This is quite different from any consumer system. Furthermore, the middleware should cover not only AV services but be usable throughout all in-vehicle Ethernet ECUs, independent from their use case, size, operating system, etc. A functioning solution for an automotive middleware that can be used in an Ethernet-based network is thus discussed in detail in Section 5.4.

5.1.4 Time-Sensitive Networking (TSN) for Safety Critical Control Data

Quality of service for Ethernet-based AV applications in the car is important. After all a very high percentage of drivers use some form of audio entertainment while driving [46]. Additionally, new driver assist functions like automated stop and go in traffic jams increase the interest in in-vehicle video entertainment. Nevertheless, the most important requirement in a car is safety. When Ethernet is used for safety critical applications – and with the prospect of autonomous driving it is likely that the high data rates Ethernet provides will be used for such applications [47] – the respective communication needs to have higher priority than any other data in the network and the arrival of the data needs to be guaranteed and that within a specific time.[14] Note that this implicitly requires a more refined distinction between different traffic classes on layer 2: not only between AV and data but also, e.g., between (time-sensitive) control traffic and other data.

An Audio Video Bridging (AVB) Ethernet network as originally developed for AV applications of "consumer devices" and "professional audio" was/is not suitable to (additionally) accommodate the time-critical control traffic in automotive (or industrial automation networks for that matter). With the completion of the AVBgen1 standards the scope of the QoS effort within IEEE 802.1 was thus broadened from AV data only to more types of data with QoS requirements and more stringent requirements. The project was renamed "Time-Sensitive Networking" (TSN) in November 2012 in order to reflect the new scope [15].

The TSN standards cater for quite a variety of requirements and degrees to which these requirements can be fulfilled. After all, the potential use cases for TSN multiply with the inclusion of control traffic and the development of new specifications. Even so (or potentially because of it), in 2016 the automotive industry was more hesitant to start any efforts to specify a respective profile on how exactly to use TSN for safety critical applications. In contrast, for the AV use automotive efforts had started quite early and resulted in a respective specification published by AVnu [38]. The following therefore just gives an overview on the different aspects the new TSN specifications address and from which the car manufacturers can choose (see, e.g., [48] for combination options). They are clustered in focus areas as proposed by [48] [49] [50].

1 TSN needed to support more use cases and with that **more data types**. It has been mentioned in Section 5.1.3.1 that the 2016 version of the IEEE 1722 specification incorporated more AV formats common in the automotive (and potentially other) industries [31]. Moreover, the new IEEE 1722 specification supports encrypted frame formats, UDP/IP encapsulation, and the tunneling of typical automotive in-vehicle networking messages from technologies like LIN, CAN (FD), FlexRay, and MOST. It also allows to distribute additional, application-dependent clock and event information, which can be useful for some use case.

2 One major concern for time-critical applications are **small latencies**. One TSN goal thus sets the maximum latency achievable to 100 μs over 5 hops. In AVBgen1 the latency goal was 7 hops in 2 ms. Concepts like the AVBgen1 credit-based shaper support the AVBgen1 requirements, but cause too much delay for TSN.

   A very basic method for reducing delays in switches is to implement cut-through switching. Cut-through switching is a proprietary method that allows an incoming packet to be sent out before the packet has been completely received. This can, in principle, be done as soon as the destination address has been recognized. However, even with cut-through switching, any incoming packet has to wait for packets currently being sent out on the same egress port to be completed, even if the priority of the currently egress blocking packet is lower. For a 100 Mbps Ethernet channel this might cause a delay of up to about 122 μs, for a 1 Gbps Ethernet channel this might take up to about 12.2 μs, at every switch on the way.[15] The following methodologies have been specified in order to reduce delays in such situations:

- The IEEE 802.3 specification on Interspersing Express Traffic (IET)\IEEE 802.3br [51] defines how in the PHY the transmission of a long, low priority frame can be intercepted and how to intersperse high priority time-sensitive traffic instead. The required "**preemption**" methodologies on layer 2 are filed under IEEE standards numbering 802.1Qbu [52]. IET and preemption can be deployed without any higher level organization in the network, as long as both ends of a link agree (e.g., with help of the Link Layer Discovery Protocol (LLDP)).

  The minimum fragment size is 64 bytes and fragments must be reassembled to its original packet before the packet can be passed onto other links in the network. The MAC merge sublayer adds a 60 bytes CRC. This means that any fragment shorter than $64 + 60 = 124$ bytes can be preempted. The worst-case delay in case of IET and preemption is thus 123 bytes, when a packet of such length was just started. Note that IET and preemption as such cannot guarantee any specific latency or delay values. They simply provide a methodology to reduce latencies for certain traffic in mixed traffic environment with long low priority frames.[16]
- **"Time-Aware Shaping" (TAS)**/IEEE 802.1Qbv restrains the original Ethernet best-effort idea yet more. It can be used in engineered networks, when time-critical information is sent at regular intervals [50]. It basically introduces a circuit-switched/TDM channel into the otherwise packet-based communication; a methodology also used in the Industrial Ethernet technology Profinet or TTEthernet [53]. With TAS, traffic with lower priority is blocked during preprogrammed, regular time windows so that the high priority control streams are not delayed by traffic with lower priorities [54]. In combination with preemption, the length of the guardband necessary before the reserved control data time window can be reduced. With TAS and cut-through switching a minimal switch latency of 1 μs can be guaranteed regardless of frame size [50].

  At the time of writing IEEE 802.1Qch was being standardized with the goal of making delays more deterministic and better determinable by emulating a cyclic transmission behavior [55].

3 For safety critical control and fail-safe operation systems, it is not only important that data arrive with small delays under normal circumstances. It is also important that they arrive at all in case of unforeseen disruptions. Two concepts are being supported by the TSN standardization efforts for this: **ingress filtering and policing** and **redundancy**, i.e., frame replication and elimination for reliability. Ingress policing prevents faulty Talkers (e.g., sensors) or switches from disrupting bandwidth and latency guarantees of other streams in network [56], when these faulty units send more data than had been assigned to them at a specific point in time. Ingress policing prevents flooding of switches at their entrance [47]. It is proposed as a fundamental mechanism to make an Ethernet network more dependable [48]. At the time of writing it was being specified in IEEE 802.1Qci [57].

For redundancy three concepts have been standardized in TSN:

- First of all, a network topology has to be designed such that **alternative paths** exist. Then central knowledge about these paths and their status has to be available. IEEE 801.1Qca then describes how to provide an alternative path in case the currently used path fails. IEEE 801.1Qca defines how to setup, modify and tear down the respective TSN streams [58].
- IEEE 802.1CB enables "**seamless redundancy**." Critical packets are duplicated and sent on alternative paths in the network. In the unit where the different copies merge back onto the same path, the duplicate arriving later is removed from the network. An added sequence number ensures that the receiver can put the packets in the right order, even if they have arrived out of order. The concept is similar to what is being done in AFDX. At the time of writing IEEE 802.1CB was being completed [59].
- Another important aspect in a TSN network is the availability of a Grandmaster clock. It therefore makes sense to ensure the availability of a Grandmaster clock also with redundancy concepts. The new IEEE 802.1AS revision provides that a single Grandmaster can transmit duplicates of its clock on alternative routes, that multiple time domains can exist, and that a redundant time master is possible [50]. In case the actual Grandmaster ceases to exist this allows for an "instant" switch over to the new Grandmaster clock. In very large networks, in which the selection of a new Grandmaster can take up to a second, this is a valuable amendment. Another element discussed for the IEEE 802.1AS extension IEEE 802.1AS-Rev is speeding up the timing and reducing the processing in a TSN network by (re)introducing a one-step clock synchronization as had originally been available in IEEE 1588 but had been omitted for IEEE 802.1AS [60].

4 Last, but not least, TSN allows for **better scalability** in the form of reduced management traffic for reservation and configuration. The respective enhanced SRP is specified in IEEE 802.1Qcc. Depending on the actual network this can further optimize the time and processing effort needed for providing QoS in an Ethernet-based network. The methodologies discussed include the use of preconfigured systems (which is opportune for Automotive Ethernet), configurable SR classes (see also Section 5.1.3.3), handling new reservations, and efficiently supporting TAS, preemption, and redundancy. At the time of writing IEEE 802.1Qcc was still being completed [61].

As can be seen, TSN offers a variety of specifications and within each specification more choices for supporting the transmission of safety critical control data within an Ethernet-based in-vehicle network. The designer of the network can choose from the TSN specifications like from a toolbox, depending on the exact requirements that need to be fulfilled [48]. The good part is, the specifications are more independent from each other than their sheer number and volume implies. Strong dependencies exist between IET and preemption (IEEE 801.1Qbu and 802.3br) and TAS requires a synchronized time (IEEE 802.1Qbv and AS). Frame preemption improves TAS (IEEE 802.1Qbu, Qbv, and 802.3.br), and redundant paths require configuration (IEEE 802.1Qca, CB).

Table 5.4 Overview of AVB and TSN specifications as provided by IEEE in respect to automotive use

	Transport	Time synch	Stream reservation	QoS	Safety (seamless redundancy)	Security
AVB (AVBgen1)	1722–2011	802.1AS-2011	802.1Qat	802.1Qav		
TSN (AVBgen2)	1722–2016	802.1AS-rev	802.1Qat 802.1Qcc 802.1Qca	802.1Qav 802.1Qbu& 802.3br (802.1Qbv) (802.1Qch) (802.1Qcv)	802.1CB 802.1Qca	802.1Qci

Note: Additional useful standards that were available prior to the AVB effort used are IEEE 1588 and 802.1Q and p.

Else, designers can make individual choices. Table 5.4 gives an overview on the AVB and TSN specification and their relation to each other.

5.2 Security and Virtual LANs (VLANs)

In the Digital Age cyber security has become a major concern. Ever since the first PC virus in the 1980s, the number of threats and attacks has continued to grow with significant (financial) impact [62]. With the increasing digitalization and increasing connectivity of cars, it is an understandable concern that also cars might become the target of hacks. Next to mere inconvenience (car cannot be used), intervention on privacy (use is monitored), and monetary losses (car was stolen, repair is needed), the potential of hacks into cars has another dimension: personal safety. A hack that succeeds in tampering with the basic driving functions like acceleration, breaking, and steering might well put lives at risk. It has been shown that the threat is real (see, e.g., [63] [64]). Attacks therefore have to be prevented as effectively as possible.

Considering the potential consequences and the deviousness of its source – after all security is needed in order to protect against the malicious criminal energy of other humans – the topic is vast (see [65] for reading recommendations). This section therefore focuses on the aspects relevant for and special to Automotive Ethernet. Section 5.2.1 structures the automotive security topic in order to be able to place security in Ethernet in the overall security context. Section 5.2.2 emphasizes on the role of switches configuration and VLANs.

5.2.1 Security in Automotive

Security in automotive first of all requires explicit consideration and an analysis of the security threats and attack surfaces on system level. A comprehensive protection

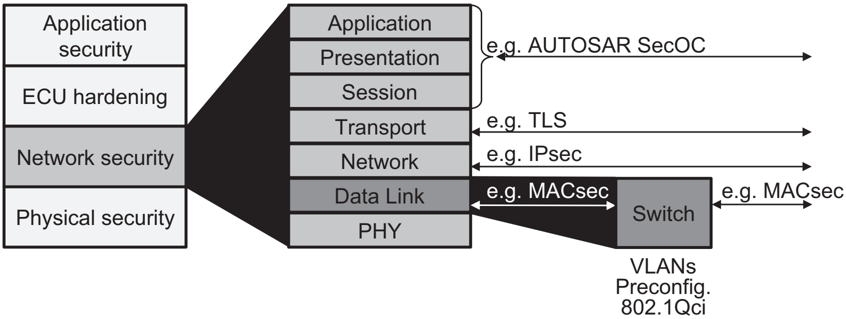

Figure 5.9 Layered automotive security approach (see, e.g., [68] [66]).

strategy is needed in order to minimize the risk of a security attack. Effective security implementations thus pursue a layered approach. The basic idea is that no implementation is perfect. Vulnerability is caused by software bugs, configuration errors, weak network design, or alike. But, if there are various layers of security and an attacker has overcome one, he still has to tackle several others. This approach is standard in the IT industry and opportune also for security in the car industry [66].

Before discussing a layered security structure in more detail, it is necessary to emphasize that (again) the circumstances in the IT industry vary from those found in the car. In IT the network is generally plug & play and unique per location. IT networks can be huge with manifold resources and frequent updates. Cars, in contrast, have a fixed topology of limited size and resources (memory, computing power, . . .). Each model is designed once and built often, with a long product life cycle and limited opportunities for (security software) updates. If one car is hacked, all cars of the same model are potential targets. Furthermore, attack patterns can change over time and become more effective. A data encryption method that is considered secure now and a car that is well protected today, might be successfully attacked in 10 years, when the same car model is still on the road. The processing power by then is likely to more easily hack longer keys that would have been too time consuming to hack today. Last but not least, an IT network is generally not restarted various times per day, and when it is restarted, it does not need to be fully operational within two seconds. A car does (see also Table 5.2). So, security methods adopted inside the car can neither use the same resources nor need as much time to complete, as IT security methods. This has to be taken into consideration when adopting security standards from the IT industry.

Figure 5.9 shows an example of what a layered security approach in automotive can look like (see left column). To start with the physical accessibility of the vehicle electronics is secured. This comprises very basic hardware measures like making it difficult to access ECUs or wiring (from the outside). Furthermore, a vehicle can be designed such that if wiring might be accessed, the communication on those wires does not connect into the vehicle network but ends at the ECU the wire is connected to. The physical security can also include architectural choices like limiting the number of ECUs with off-board communication.

www.cargeek.ir

The next level represents the security on the networking level, which again can be divided into layers, i.e., measures effective on each ISO/OSI layer. The depiction in Figure 5.9 includes examples of existing authentication and encryption protocols that have been developed by the IT or automotive industry and that implementers can choose from. The following gives a brief overview only as the emphasis of this section is on security measures inherent in the Ethernet technology, that have nothing to do with cryptography (see Section 5.2).

- The AUTOSAR SECure On-board Communication (**AUTOSAR SecOC)** has been developed in order to provide a resource-efficient and practical security mechanism that seamlessly integrates into the AUTOSAR communication [67] and that, being on AUTOSAR level, can be used with all networking technologies supported in AUTOSAR (CAN (FD), FlexRay, Ethernet, LIN). It provides end-to-end authentication and integrity based on message authentication codes and freshness values (counter or timestamp). For efficiency in computation and bandwidth consumption it assumes symmetric keys [68], though asymmetric keys are not precluded.
- A typical security attack on TCP level would be a TCP sequence prediction attack. Knowing the packet sequence allows to send counterfeit packets, which can potentially harm the receiver. A protocol used to prevent such attacks is the **Transport Layer Security (TLS)** protocol (previously known as Secure Sockets Layer (SSL)). This protocol ensures privacy and data integrity between two communicating applications like HTTP, IMAP, SMTP etc., by providing encryption and authentication mechanisms [69]. The protocol supports a number of different methods for encryption, cryptographic key exchange, and authentication, which are negotiated between client and server. TLS 1.2 was specified in the RFC 5246 in 2008 [70], with TLS 1.3 being work in progress in 2016 [69]. TLS is used with TCP and does not work with UDP.
- The basic concern in an IP network is that every router an IP packet passes through can read the packet and even change its content. It is also possible that one node could send a packet pretending to be another node, by using that node's address in the origination address field (called IP-spoofing). In this context the Internet Protocol SECurity (**IPsec**) was developed. Its goal is to ensure end-to-end privacy, authenticity, and integrity across the, in principle, not secure Internet. IPsec uses various mechanisms to achieve this like encryption and the addition of a header element containing a message authentication code; all directly integrated on layer 3 of the ISO/OSI layering model and thus transparent for the applications [71]. IPsec was developed in conjunction with IPv6, but can be and is used with IPv4 as well. It was standardized by the IETF in a number of RFCs (see [72] for an overview). IPsec covers more corner cases than MACsec (see below). IPsec AH can be used if authentication is of interest only and can be implemented without much effort.
- A typical attack on the MAC layer is ARP spoofing. The Address Resolution Protocol (ARP) resolves IP addresses into MAC addresses. In an attack, the attacker sends a message with the IP address of a masqueraded host, but with its own MAC address. The receiving node caches the falsified IP and MAC address combination.

The attacker is thus in a position to intercept, to manipulate or interrupt the communication and can start other attacks such as flooding/denial of service and paralyze the complete network. The IEEE 802.1 thus standardized IEEE 802.1AE,[17] whose latest amendment, 802.1AEbw, was released in 2013 [73] and which is generally referred to as MACsec. MACsec offers P2P encryption and authentication between directly connected nodes. It is performed for every hop also protecting the VLAN tag and not end-to-end like IPsec or SecOC. Especially its authentication algorithm is of interest for the automotive industry, as it provides a good level of integrity in the single infrastructure with mixed security domains that the in-vehicle network represents. However, to be usable in automotive, MACsec requires, like many of the TSN features discussed in Section 5.1.4, hardware support in controllers and switches and thus adds costs to the semiconductors. At the time of writing the discussion in the automotive industry on methods to use for Ethernet-based communication had not be concluded [74].

Cryptography is obviously very important for network security. There is a variety of algorithms available for different purposes like key exchange, peer authentication, message authentication, message encryption etc. Their details are not decisive in the scope of this book and will therefore not be discussed (see [75] for suggestions on respective publications). Crucial in automotive is their implementation. It needs to be effective and fast. In comparison with other in-vehicle networking technologies, the implementations need to cope with significantly higher data rates when used with Ethernet. Pure software-based implementations are not efficient enough, neither in terms of processing time nor in the use of resources. Hardware support is therefore needed, which is typically provided with help of a dedicated Hardware Security Module (HSM). An HSM efficiently executes cryptographic functions and securely stores cryptographic keys [76]. In consequence, the implementation of crypto-algorithms offers opportunities for suppliers to differentiate their products and is not discussed further here.

Once the communication in the network has been secured, the ECU itself, i.e., its software and electronics, need to be protected. This concerns the ECU implementation, the processors chosen, the partitioning of software on the processor or onto different processors and alike. Also, ports, currently not active can be deactivated in order to protect the ECU better. On the highest level, the application can comprise, e.g., plausibility checks and data use policies additional to yet more authentication and encryption. An application can be made to accept expected data only or to accept certain data, like control messages, only in specific application states. Anomalies can be detected if, e.g., cyclic messages are received more often than defined or if sensor data contains undefined information.

Naturally, a layered security approach makes sense also in case of other in-vehicle networking technologies. The approach as such is not Ethernet specific. However, this section shows that Ethernet-based communication in automotive benefits from the 15+ years head start, the IT industry has in security compared with the automotive industry [77]. The notion that introducing Automotive Ethernet weakens the security in cars is thus incorrect.

Note that at the time of writing, the industry had not only not yet converged on algorithms to use for security with Automotive Ethernet but there also was no respective industry-wide standardization activity. Various efforts existed that discussed other, specific aspects of security in automotive. One of the outcomes has been discussed above, the AUTOSAR secOC [67]. Another outcome was produced by the US based Society of Automotive Engineers (SAE), which was working on recommended practices for in-vehicle cybersecurity [78]. The focus of the resulting SAE J3061 is on the integration of cyber security in automotive processes and not on specific protection mechanisms. Furthermore, members of the German Association of the Automotive Industry (VDA) had initiated the ISO 21434 standardization project in order to cover procedural aspects of providing automotive security [79] and in the Japanese car industry, JASPAR had set up a security group in their organization [80].

5.2.2 Ethernet-Specific Security Aspects

In an Ethernet network there are two properties of importance that can have impact on the security: (a) communication can be broadcast communication and (b) there is no default control in an Ethernet network on how much traffic a network participant can sent [81] [74]. Broadcast happens with all unrestricted broadcast and multicast messages, but also when the addressee of a unicast message is not (yet) known. So broadcast can happen any time. For the ECUs, in theory, they should be designed such that they do not send too much traffic. However, next to design errors or simple malfunctioning that might cause a unit to send too much data, this can be just the outcome of a malicious attack security measures want to prevent.

Both, too much broadcast and too much traffic being sent by even only one participant, can flood the network on MAC level and result in a denial of service or other malfunctions of the network. Additionally, broadcasted messages as such can be listened to anywhere in the network. The following thus investigates the means available on switch level that support these two tasks: 1. Stop too much traffic from coming into a switch and 2. Stop too much traffic from going out (see also Table 5.5).

Section 5.1 discussed different methods available with AVB/TSN. Noteworthy in the context of security are the Stream Reservation Protocol (SRP, see Section 5.1.3.3) and ingress filtering and policing/IEEE 802.1Qci (see also Section 5.1.4). As the overview in Table 5.5 shows, ingress policing is the only really suitable means to prevent too much data at the ingress of a switch. On the other hand, the SRP has some but only small influence on the traffic at the exit ports, because the SRP switches impose transmit limits at class level and not for individual streams, senders, or receivers. The following two subsection explain the effects that can be achieved when using VLANs and when using the switch configuration accordingly. Last but not least, Section 5.2.2.3 will briefly cover the topic of key management.

5.2.2.1 VLANs

One way to structure data within an Ethernet network is the differentiation between infotainment, control, and best-effort traffic, as described in Section 5.1. Another way to

Table 5.5 Overview of means to mitigate security threats in the Ethernet switch (without authentication or encapsulation)

	AVB/TSN	VLANs	Switch configuration
Stop too much traffic from coming in	**(++)** Ingress policing, IEEE 802.1Qci	**(+)** VLAN filtering can, e.g., drop packets with no, unknown, or unsupported VLAN ID	
Stop too much traffic from going out	**(+)** SRP limits the outgoing traffic per class and port	**(++)** limit VLAN traffic to respective VLAN **(+)** VLAN broadcast zones **(+)** Only forward packets with VLAN tag **(0)** add VLAN tag to packets without	**(++)** (semi)static ARP and MAC address forwarding tables **(++)** multicast filtering, define rules for packets with unknown addresses

Note: For authentication and encapsulation, see Section 5.2.1.

structure the data is by assigning data to different Virtual LANs (VLANs) as defined in IEEE 802.1Q. A VLAN describes a virtual Ethernet segment in the Ethernet network, in which all participants are identified by the same VLAN ID (see also Figure 5.3 or Figure 1.5 in Section 1.2.1). This means that a VLAN enabled switch will pass on packets with a VLAN ID only between units of the same VLAN and not to others, even if units of other VLANs are physically connected to the same switch or if the message is a broadcast message. VLANs were developed in the context of significantly larger IT networks, in order to be able to handle and segment them, but they are useful also in the context of Automotive Ethernet.

In respect to security, VLANs can thus be used to isolate traffic and to reduce broadcast zones. Depending on the design, the isolation can be between critical/uncritical traffic, internal/external traffic or it can isolate the traffic flows of different application areas or security zones. An example of separating different VLAN traffic flows is shown in Figure 5.10 and described further below. VLANs can also be used to perform some ingress policing and drop all packets that are not part of the VLANs the switch supports. If packets arrive without VLAN tag, the switch can drop the packet or it can add a tag based on available packet information like port, protocol, fixed header fields etc. [74]. To manipulate this would require hacking into the switch configuration interface, which often is a host controller or µC. Consequently, in case of VLANS it is the µC that would need to be hacked, which should in any case receive considerable security protection.

Note that next to improving the security, a smart use of VLANs can simplify various challenges. Examples are:

- **Data logging and testing:** VLANs give flexibility in relating ECUs to network segments, independent from the physical location of the units. This will have an increasing importance for data logging and analysis in growing Automotive Ethernet networks.

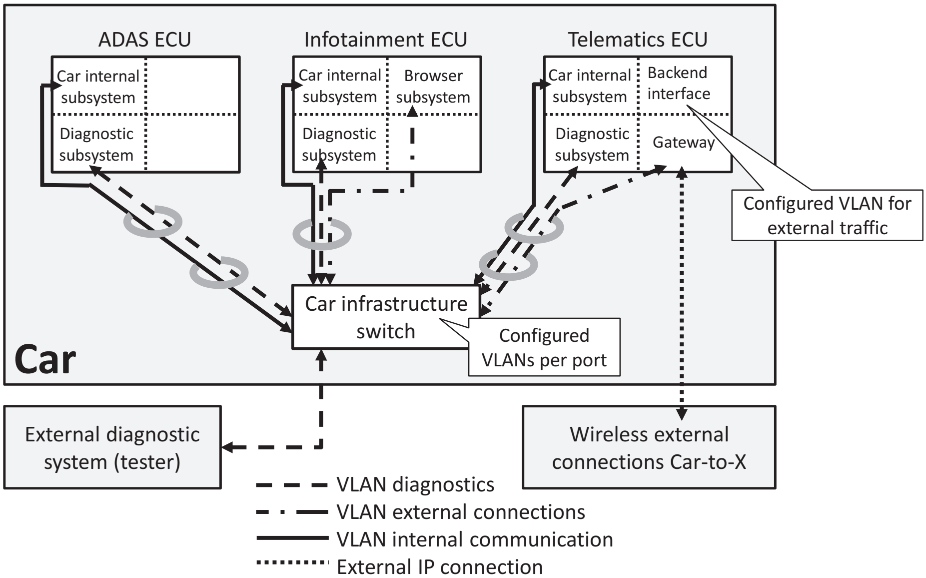

Figure 5.10 Example use of VLANs including software compartmentalization.

- **Performance:** A certain communication can be assigned to a specific VLAN and this VLAN can be prioritized within the switches.

Figure 5.10 shows an example of what traffic isolation in an in-vehicle Ethernet network could look like. In the example an ECU called "Car infrastructure switch" has been selected as the major unit to perform traffic isolation. This unit can physically located anywhere inside the car. The VLAN filtering rules, defined during the development process, are applied per port. The depiction also shows where adding or removing a VLAN tag makes sense: for both external interfaces. When, e.g., diagnosing the car via the standardized diagnostic interface, the diagnostic traffic simply receives the respective diagnostics (VLAN) tag and is then distributed in the diagnostics VLAN inside the car. Within the car, this traffic is always seen as "diagnostics traffic" and never as "car internal traffic," which makes effective isolation quite easy and results in a quite efficient firewall. The external tester is unaffected.

The situation is the same for the other external interfaces most modern cars will have, e.g., for connecting to the Internet. In this case, it is possible to work identically. Traffic that enters or leaves the car via one of the many radio interfaces can be tagged as coming from outside, while the tag is removed when data from the car leave via the same interface. Inside the ECU such tagged traffic is handled in an isolated area only. This ensures that a browser application has no access to car internal data, even if it passes along the same wires. The strict separation of traffic is crucial and car manufacturers should ensure this with help of a respective development processes.

VLANs are unprecedented in cars. Their implementation offers a powerful tool to the designers of the Automotive Ethernet network and will also provide competitive advantages for those who do it well. It is therefore unlikely that the automotive industry will standardize how to implement VLANs on a general basis. However, for specific use cases, like the automotive camera interface (ISO 17215), the use of VLANs is addressed.

5.2.2.2 Other Switch Configuration Possibilities

The main task of a switch is to look at the address fields of a packet received and to forward it to the exit port(s) behind which the destination address can be found. In order to be able to do this, the switch maintains a forwarding table. This table is being filled by the switch remembering from which port packets with which source MAC address comes from, so if the same source address is found later in the destination MAC address field of a different packet, the switch knows behind which port the unit can be found. If a destination address is not yet known, the switch forwards the packets to all ports (i.e., broadcasts it). A potential attacker could thus flood a network with broadcasted messages by continuously sending packets with destination addresses unknown to the switch.

Such attacks can be prevented if the MAC address learning of the switch is limited to an initial starting/setup phase using the first packets only or if the address learning is switched off completely and instead a preconfigured static forwarding table is used. This includes the possibility that the forwarding table is learned once with the first start in the factory and that the resulting forwarding table is stored and used after that. The same applies for the ARP tables that match MAC and IP addresses. In the static network of a car these are viable and advisable proceedings. Naturally, the learning can also simply be limited based on number of entries, address range, and frequency of change [81].

Additional filtering rules can be defined for multicast messages. Without filtering rules, multicast messages are also simply broadcasted in the network. Rules that match specific multicast addresses to a fixed set of destination addresses can ensure that the switch forwards multicast messages only to those. Furthermore, it should be defined upfront what to do in case multi- or unicast messages are received with unknown destination addresses. Such messages can be dropped, or likewise forwarded to (a) specific address(es) only.

References [81] [74] propose the use of Access Control Lists (ACLs) to precisely configure packet forwarding in a switch. ACLs have originally been developed for IP routers for applying forwarding rules on layer three, but they can also be found in switches. An ACL is a list of match-action pairs that can be applied to VLANs or ports and typically permit or deny transmission based on the bitwise match of the Ethernet or higher layer protocol headers (e.g., IP, UDP, or TCP). Table 5.5 gives an overview on the different, Ethernet inherent means and their effectiveness to mitigate threats of too much traffic entering or leaving the switch.

Naturally, the switch can be used to enforce authentication such that switch ports are deactivated and will only be activated for normal communication after the connected node has been successfully authenticated. The authentication of the network nodes need to be done by the switch firmware or the microcontroller directly connected to the switch to guarantee a fast start-up.

5.2.2.3 Efficient Key Management

As a basic principle of cryptography the same key should not be used for different functions and devices and a key should not be used for a long period of time [82]. Additionally, data authentication, key exchange, and data encryption each requires a different key. If the key for data encryption is compromised, for example, the other keys are not affected and a new key can be assigned via the existing encrypted key exchange. To limit the amount of data for which a key is used, every key is assigned a specific lifetime for which it is valid, depending on its use. Moreover, communication can be divided into separate groups with separate connection keys according to the respective vehicle domain or other functional aspects.

The distribution of the keys to the ECUs in the vehicle is a very complex task. The usual methods employed in the IT world, such as the Internet Key Exchange Protocol IKEv2 (IETF 4306) and X.509 certificates (IETF 5280), are not suitable for in-vehicle key management. Those methods would consume too many resources and an online connection to a Certification Authority Server would be required. This can neither be guaranteed at all times nor is it fast enough. Instead, one ECU in the vehicle needs to assume the role of a key master which distributes the keys to the other ECUs.

A potential solution is the following: The key exchange is realized by means of symmetric encryption and is triggered through a diagnosis request, an elapsed period of time or from a service backend server outside the vehicle. In this setup the key master is the only ECU that communicates with the service backend server regarding key management. It uses asymmetric cryptography methods for this purpose.

5.3 The Internet Protocol (IP)

The Internet Protocol (IP) is the fundamental protocol associated with enabling and using the Internet. However, it is only one of several protocols that represent the "Internet protocol suite" or "TCP/IP protocol suite," a combination of protocols that enables vendor and operating system–independent communication between networking enabled electronic devices [83].[18] The first version of the protocol(s) was published in 1974 as RFC 675, "Specification of Internet Transmission Control Protocol" (TCP) [84]. In 1981, with the fourth version, TCP and IP were, for the first time, separately described in RFC 793 [85] and RFC 791 [86]. The User Datagram Protocol (UDP) was standardized in 1980 (RFC 768 [87]) and is also part of the TCP/IP protocol suite. Today, most networking uses the TCP/IP protocol suite. Thus, cars have to facilitate it if they are to be handled as nodes in the (worldwide) network. Cars also have to support it as part of the Ethernet-based communication in in-vehicle networking. In short, the TCP/IP protocol suite is a fundamental part of AANs.

In general, there are no specific automotive requirements when implementing the respective protocols. After all, the possibility of reusing standard compliant implementations of protocols like TCP, UDP, and IP is one of the reasons for doing Ethernet-based communication in the first place. One aspect that has to be taken into account when implementing the TCP/IP protocol suite is the footprint of the software. Small ECUs,

Table 5.6 Original IPv4 addressing classes

Class	Range class	No. networks	Hosts per network	Range private	Amount private	Purpose
A	0.0.0.0 to 127.255.255.255	126	~16.78 Mio	10.0.0.0 to 10.255.255.255	~16.78 Mio	Unicast
B	128.0.0.0 to 191.255.255.255	16.384	65.534	172.16.0.0 to 172.31.255.255	~1.05 Mio	Unicast
C	192.0.0.0 to 223.255.255.255	~2.1 Mio	254	192.168.0.0 to 192.168.255.255	~0.066 Mio	Unicast
D	224.0.0.0 to 239.255.255.255	n/a	~221 Mio	n/a	n/a	Multicast addresses
E	240.0.0.0 to 255.255.255.255	n/a	~315 Mio	n/a	n/a	Reserved for experiments

Note: The use of A, B, and C has been obliterated with CIDR and, of course, IPv6.

like cameras integrated into the side mirror, have little processing power available in optimized DSPs or μCs. It is therefore important to pay attention to the implementation of the software [88]. The skillful implementation of the TCP/IP stack gives a competitive advantage to those capable. However, it is not the topic of this book.

The portion of the TCP/IP protocol suite that does leave some structural choices in automotive is the use of IP. The core functions of IP are addressing, i.e., identifying and locating units (called "hosts" in IP), and routing packets from a source address to a destination across, well anything, from within a small network to across multiple networks and around the world. In the public Internet, IP addresses thus have to be globally unique. To ensure this, Regional Internet Registries distribute the IP addresses, while the Internet Assigned Numbers Authority (IANA) publishes the availability of addresses. Additionally, there are IP address ranges available for closed/private networks (see, e.g., Table 5.6). Anybody can use these, as long as the units using them have no access to the global network. Examples for closed networks are factory floors, on which robots communicate (only) with each other or, in parts, cars (see Section 5.3.1 for more details).

An Internet Protocol version 4 (IPv4) address consists of 32 bits, which originally only identified the network and the host in the traffic classes shown in Table 5.6. It became evident quickly that the original concept was not sustainable and various methods were developed either to make routing more efficient and/or to make the address range stretch further. Examples are subnetting, Variable Length Subnet Mask (VLSM), and Classless Inter-Domain Routing (CIDR) [89].[19] Despite these efforts, the predominantly used IPv4 has run out of addresses [90] and everyone designing networks using IP today – which includes the automotive industry – has to integrate the use of IPv6 (see Section 5.3.2). IPv6, first published in 1995 [91], uses 128 instead of 32 bits for the address, so there is hope that the address space will last longer.

Last, but not least, the question of security is often discussed in the context of IP. Because IP is the connecting element in the worldwide network, it is seen as an entry point also for undesirable elements. Section 5.2.1 provides some basic considerations for security in automotive, including a brief description on IPsec, a security protocol to use on layer three.

5.3.1 Dynamic versus Static Addressing

Modern IT systems need to be very flexible. They have a changing number of nodes and routes in the network. One of the key elements to support this is the dynamic setting of IP addresses. Deploying Dynamic Host Configuration Protocol (DHCP) and Domain Name System (DNS) servers is state of the art. Automotive Ethernet networks are entirely different. An in-vehicle network is an almost closed system. Even if the number of active nodes in an in-vehicle network may vary (for more details see Sections 5.4.4 and 6.3.3), the maximum number of nodes is more known and more limited upfront. Furthermore, in contrast to IT networks, cars might be started and parked several times a day (see also Section 6.3.1). Fast start-up is therefore very important, which makes a static IP configuration the natural choice. However, there are some use cases in which dynamic IP addressing makes sense in cars, too.

The following possibilities exist to assign IP addresses inside cars. The list shows how Automotive Ethernet adds, with IP addressing (and VLANs, see Section 5.2.2.1), another design dimension to the electronics development inside cars.

- **Static:** The IP addresses are assigned during the development and every ECU of the same class, e.g., Head Units (HU), always receives the same address, independent from the car they are built into. As the same address is obviously used repeatedly between cars, it is selected from a specific address range. The private address pool listed in Table 5.6 would be a designated source for it, but does not have to be. In the case of static addressing it is absolutely important that never two ECUs of the same class are built into one vehicle.
- **Pseudo-dynamic (branding):** In this case the ECU is delivered without IP address, but receives a then static IP address during the assembly. Consequently, after the address has been assigned, it cannot be changed anymore. This process is needed in case the same part is assembled multiple times inside a car. The cameras of the surround view system provide an example. Exactly the same camera is placed at different locations inside the car. So, for assembly and also for repair, the cameras are delivered without IP address. This reduces costs for logistics and storage. This branding procedure is standardized with the automotive camera interface of ISO 17215 (Part 4).
- **Dynamic:** This is required, when the vehicle or parts of the vehicle communicate with external components/the world outside the vehicle, e.g., the diagnostic tester (see ISO 14300) or the Internet. In this case, it is no longer possible to use addresses from the car internal address space. Instead, the ECUs directly connecting the car to the outside world, "port ECUs," use the IP address(es) an externally located DHCP server assigns to them. One possibility to connect other ECUs inside the car via those "port ECUs" is to implement a dynamic/static address translation with a Network

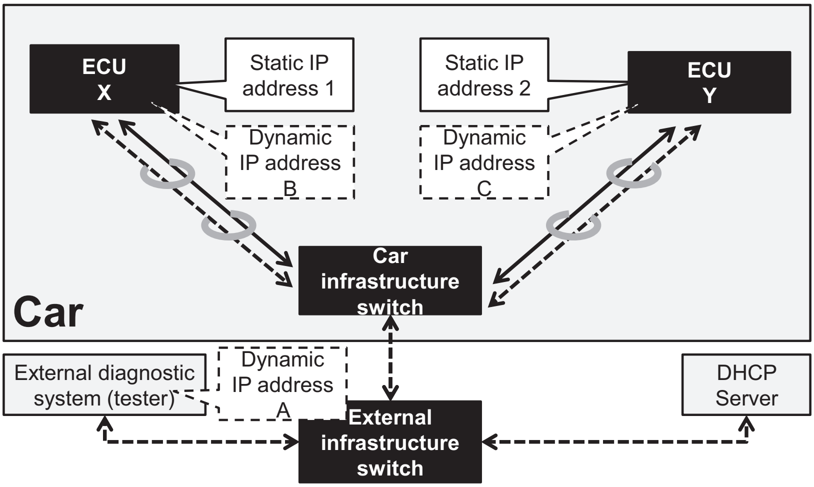

Figure 5.11 Example use case for multiple IP addresses (solid line for regular vehicle internal traffic, dashed line for diagnostic traffic).

Address Translation (NAT) in the port ECUs. Another is to request more temporary IP addresses from the external DHCP server.

- **Multiple:** In this case, an ECU accommodates and uses several IP addresses. This is the case if ECUs want to use various address spaces. One example is the diagnostics use case (see Figure 5.11). In this case the car internal network uses static IP addresses. The external tester cannot know the internal address structure, and the internal network cannot participate in the external communication. For the time of the testing, additional IP addresses are thus assigned by the DHCP functionality of the external network that the tester is part of. Figure 5.11 shows an example of the use of multiple IP addresses. As can be seen, where appropriate, separation of traffic can be achieved also with multiple IP addresses assigned to the communication partners on layer 3, and not only with VLANs on layer 2. In the example shown, only the dynamic addresses B and C are needed and they are thus assigned only when the diagnostic system is connected and would like to communicate with the internal components directly.

5.3.2 IPv4 versus IPv6

With the address space of IPv4 running out [90], migration scenarios from IPv4 to IPv6 are often discussed [92]. However, this is not really a concern for Automotive Ethernet nor does the automotive industry face challenges that are different from those of other use cases. A significant amount of automotive communication is internal to the vehicle only and it is the explicit intent that there is no interface to the outside world.

This communication can continue to use static IPv4 addressing, if this is desired, e.g., because the complexity is smaller than for IPv6. For the communication between car and external world, this is of course different. The car has to integrate into the network the outside world defines and the respective components naturally need to support IPv6 to be future-proof. With the multiple addressing scheme an Automotive Ethernet network can support both.

5.4 Middleware and SOME/IP

5.4.1 Definition of "Middleware"

This section starts with the disambiguation of the term "middleware." The term originates in the development of complex software systems and addresses all functions that are needed for a "service" to allow data exchange between otherwise decoupled software components. This data exchange often passes through a network and it is the task of the middleware to ensure that the network used is transparent to the software components exchanging the data. As is shown in Figure 5.1, middleware ("SOME/IP") cooperates at the higher layers of the ISO/OSI layer model. It organizes the transport of complex data (messaging) and moderates function calls (Remote Procedure Calls, RPCs) between the software components.

One of the disadvantages of using a middleware is its size and load. However, with the increasing size and performance of software systems this is mitigated. Furthermore, it becomes ever more important to be able to handle complex software systems comfortably and to improve their quality; with middleware being a suitable tool for it. The advantages of using modern middleware are the improvement of distributed development on different software modules and the much better testability of the modules.

The amount of software in cars today can be huge [93] and is still increasing along with the distribution of functions and systems inside cars (see also Chapter 7 for an outlook). These distributed functions can use various processes within one ECU but they might be spread also over various processes in different ECUs. With the thus increasing complexity, simply placing messages onto the network under the assumption that the correct function will receive them is no longer sufficient. RPCs are required to control the distributed functions and the correct methods to initiate this. Additionally, different ECUs might use different software architectures (and Operating Systems, OSs). This means that middleware also has the important role of bridging between Portable Operating System Interface (POSIX) capable Unix-like operating systems such as Linux or QNX and AUTOSAR systems, which are all used in automotive.

5.4.2 The History of SOME/IP

When starting the development of Automotive Ethernet, the intention was to reuse one of the many middleware solutions available; preferably one following an open source licensing model. Various approaches were scanned and some solutions, such as Etch [94] [95] [96] or Google Protocol Buffers [97] (serialization only) for middleware or Bonjour [98] for Service Discovery (SD), were investigated more closely. In principle,

www.cargeek.ir

both solutions could have been modified to fit the small processing capacities available. However, two issues remained unsolved:

- **AUTOSAR** provides many software modules, which incorporate some of the middleware functions and are configured with the help of a separate tool chain. In order to avoid **incompatibilities**, the reuse of existing (IT) middleware solutions would have required either bypassing the AUTOSAR modules or ensuring the use of the same data types and the partitioning of the existing middleware solutions such that they could have been integrated in AUTOSAR.
- **The licensing of the existing (IT)** middleware solutions was **not** quite **as needed**. Although the licensing of the respective implementations of the investigated solutions was open, the essential patents needed for adapting the solutions were not. Instead, those patents were protected and owned by large IT companies, with unknown consequences.

While, in theory, it would have been possible to make one of the technically suitable solutions usable with AUTOSAR, this was not possible in combination with the licensing issue. It was thus decided to develop a new solution. To reduce the risk of running into licensing issues with the new solution, the IPR situation was taken into consideration, while at the same time the new solution was being published as state-of-the-art technology. Naturally, the solution was developed to be directly usable with AUTOSAR systems. The Scalable service-Oriented MiddlewarE over IP (SOME/IP) specification has thus been an integral part of AUTOSAR since AUTOSAR version 4.1. Additionally, SOME/IP is provided as a GENIVI library. More public information is available on [99].

5.4.3 SOME/IP Features

SOME/IP was designed to support the following features needed for the automotive use cases:

- Service-based communication approach
- Small footprint
- Compatibility with AUTOSAR (no other middleware is AUTOSAR compliant)
- Scalability for the use on very small to very large platforms
- Flexibility in respect to different operating systems used in automotive like AUTOSAR, OSEK, QNX, and Linux

5.4.3.1 The Header Format

Figure 5.12 shows the SOME/IP header. The individual elements are explained in the text below.

- **Message ID:** The first 16 bits of the Message ID identify the service (Service ID) used. The service provides the overall structure for the middleware communication. An example of a service could be "*CD_Player*" (the complete example is given in the notes[20]). Each service needs to have a unique Service ID, which the system integrator

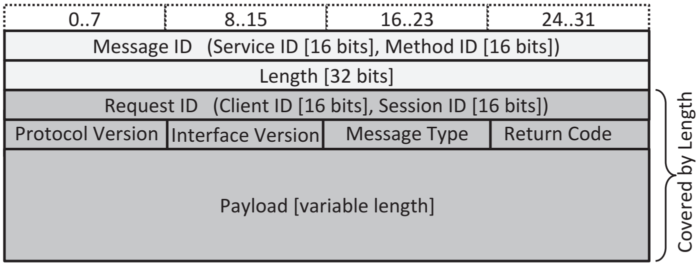

Figure 5.12 Header format for SOME/IP.

assigns. A service can consist of a set of methods, events, and fields, which are identified in the Method ID. The 16 bits for the Method ID represent the other half of the Message ID. An example of a method could be "track_number.set." In comparison, CAN provides only a small subset of what is possible with service-based communication. However, the idea behind the SOME/IP message IDs is similar to that of CAN message IDs (see Section 2.2.2.2). It is therefore possible to treat the SOME/IP message IDs with the same process structure, which just needs to be enhanced/adopted for SOME/IP.

- **Length:** The length field uses 32 bits to specify the number of bytes including the payload, some header information, and the Request/Client ID (see Figure 5.12).
- **Request ID:** The Request ID allows a client to differentiate between multiple calls of the same method. The first 16 bits of the Request ID are called Client ID and identify a specific client. For example, if a user would like to set the track in the CD-player (server) from the Head Unit (Client A), this would have a different Client ID than if a user of the Rear Seat Entertainment (RSE) (Client B) would like to set the track in the same CD-player. The second 16 bits of the Request ID represent the Session ID. If, e.g., Client A sends the message to set the track in the CD-player multiple times, each of these messages receives a different Session ID. When generating a response message, the server always has to copy the Request ID from the request to the response message. This allows the client to map a response to the correct request. The Request ID is an inheritance from AUTOSAR's Client/Server communication.
- **Protocol Version:** An 8 bit field which identifies the SOME/IP protocol version. At the time of writing, SOME/IP has the version 1.
- **Interface Version:** These 8 bits identify the major-version of the service interface. The interface definition and version numbering is up to the designer. In case additions are made and new versions are defined, this field in the header allows the automatic detection of version incompatibilities in the design.
- **Message Type:** This field differentiates between the different possible types of messages. With SOME/IP version 1.0 the values shown in Table 5.7 were defined.
- **Return Code:** The 8 bits of the Return Code signal whether a request was successfully processed.
- **Payload:** The payload field contains the parameters of the SOME/IP message. In the case of the example, this might be "10," if that represents the value the track should

Table 5.7 Important SOME/IP message types

Value	Name	Purpose
0x00	REQUEST	Request expecting a response (even void)
0x01	REQUEST_NO_RETURN	Fire and forget request
0x02	NOTIFICATION	Request for a notification (i.e., a subscription for an event call back or a field value) expecting no response
0x80	RESPONSE	The response message (for a REQUEST or as a result of a NOTIFICATION)
0x81	ERROR	In case a RESPONSE cannot be delivered because of an error

be set to. The size of the SOME/IP payload field depends on the transport protocol used. For UDP, the SOME/IP payload can contain 0 to 1400 bytes. The decision to limit the payload length to 1400 bytes was taken in order to allow for future changes to the protocol stack, like using IPv6 or adding security protocols. Since TCP supports the segmentation of payloads, larger payload sizes are automatically supported. With the SOME/IP Transport Protocol (TP) segmentation larger payload sizes are also supported for UDP. The serialization of parameters, i.e., the order of the values in the payload and in which order to place the least to most significant bits, is also specified in SOME/IP.

5.4.3.2 The Service Concept and the Supported RPC Mechanisms

SOME/IP defines a service by its Service Interface, i.e., the activities of client and server, based on the defined communication principles. In this, a Service Interface is comparable to a MOST FBlock (see also Section 2.2.4.2). Figure 5.13 gives an overview

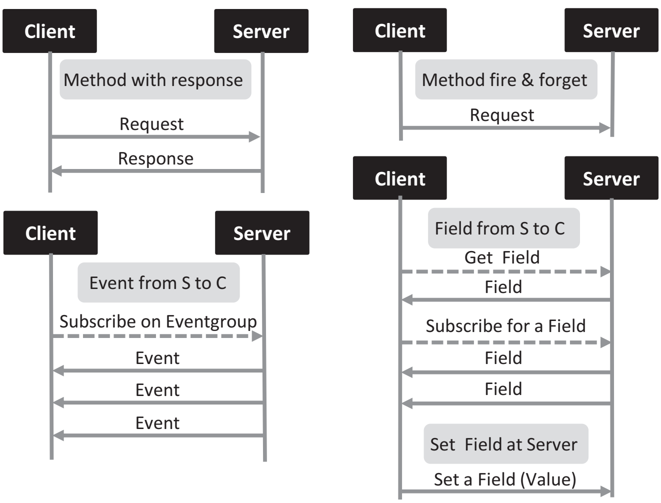

Figure 5.13 Communication principles supported by SOME/IP. C = client; S = server.

of the different communication principles supported by SOME/IP. A Service Interface may include (a) methods with response (request/response) or without response (fire & forget); (b) events, i.e., a message from the server to the client when something happens; (c) fields, which get, set, or notify of a property or status; or (d) event groups, which are logical groups of events and fields used for publish/subscribe handling. Figure 5.13 visualizes these basic communication principles.

- **Request/Response:** Describes a method with Request and Response messages. The Request is a message from the client to the server calling a method. The Response is a message from the server to the client transporting the result of the method invocation.
- **Fire & Forget:** Describes a method with just Request messages. Like in the Request/Response case the client invokes a method at the server. In contrast to the Request/Response case, the client does not expect a Response.
- **Events:** In this case the server sends messages with specific information to the client, either cyclically or when there is a change (event). Prior to this, the client will have told the server that it wants to receive the information, i.e., will have "subscribed." As the server expects no response from the client, this could be also seen as a "Fire & Forget" communication principle from the server side. The Event messages are similar to regular CAN messages.
- **Fields:** Fields represent "properties" that can be accessed remotely. The communication principles for "get" Fields are in line with Events. In addition, Fields can be "set" by the client. Furthermore, the "properties" are available at all time that system is alive, while the Event is only valid within the time this event is happening. Thus, a "property" can be seen as a kind of software variable that can be addressed from an external interface. Fields are similar to "Properties" in MOST.

5.4.4 Service Discovery (SD)

Automotive Ethernet is often just seen as a new in-vehicle networking technology that simply allows for higher data rates. However, there are a few more important differences. In the context of the communication methods, the key difference is that Ethernet-based communication provides for service-orientation. Until the introduction of Automotive Ethernet, MOST was the only in-vehicle networking technology supporting a service-based approach, and therefore service-oriented communication was only deployed in the infotainment domains of those car manufacturers using MOST. That MOST/infotainment uses a service-based approach is not by accident though. High-end infotainment systems were the first to need the more complex interfaces that service-orientation provides. Examples include complex data types, access to databases, the transmission of lists, etc. Also the use of Remote Procedure Calls (RPCs) was first required in infotainment and is thus also supported by MOST.

In the rest of the in-vehicle domains the "CAN-approach" dominated, i.e., information is put onto the channel and it depends on the mechanisms implemented in the receiver if and how that information is processed. However, especially in innovative user domains like the driver assist domain, the "CAN-approach" is becoming less suitable to cover the communication requirements. Additionally, with a data payload of 8 bytes, CAN does not allow for extensive header information, which limits the adoption of new concepts like RPC or SD. With Automotive Ethernet being used in the driver assist domain and more service-orientation diffusing into other in-vehicle areas, the authors are convinced that a general shift to service-oriented communication is necessary to meet the in-vehicle communication requirements of the future.

Service-based communication is one of the key properties and key advantages of SOME/IP. Also important is the addressing. Communication is not only via broadcast but uses unicast as well. With unicast, it makes sense to address communication partners only if they are really available.

With SOME/IP-SD, SOME/IP also supports a mechanism that determines which and if a service is available or not. Service Discovery (SD) as such, however, is still a controversial subject in the automotive industry. The main counterargument is that the in-vehicle network and the availability of functions are not dynamic enough to justify the use of SD. The following list reflects a number of situations in which the seemingly static network is faced with increasing dynamics, but for which SD provides a solution:

- **Dynamics during start-up:** One of the more complex tasks in the system design of cars is the start-up. Each ECU in a car can show a different start-up behavior. Some ECUs will start-up quickly; others will be slower. Some ECUs start even if the voltage drops down to 3.5 V; for others a start-up voltage of 8 V is not sufficient. This means that during the start-up, functions are available at different points in time. Without SD, a hard limit needs to be set that defines the point in time at which all functions are expected to be available. It needs to be defined according to the function or ECU needing the longest start-up time. With the above mentioned different option/combinations, the time limit would either be different for every vehicle or always the longest. In contrast, with SD available, every function/ECU can announce its availability when ready and, in general, user functions can be available earlier. This significantly simplifies the process. During start-up, SD has another advantageous side effect in a switched Ethernet network: The switches can learn the addressing tables directly with the SD messages.
- **Dynamics because of customer variants:** Most car manufacturers offer options for their customers to choose from when buying a car. As a rule of thumb, the bigger and more premium a car is, the greater the number of options that can be selected. A large number of options means an even larger number of combinations of options, so in consequence many car manufacturers build individual cars according to the specific customer's requirements. Without SD, each ECU needs to be statically configured with respect to the availability of functions of other ECUs in the car. With SD available, ECUs can establish on their own which other functions/ECUs are available in a vehicle, without requiring any option combination–specific preconfiguration. This is significantly less error prone. Therefore, the more complex the car, the larger the advantages of SD are.
- **Dynamics in the event of failures:** In a network that functions with Fire & Forget messages only, it is not always directly evident when a communication partner has

failed. An ECU not detecting any related messages on the link might simply assume that a certain event has not happened or that a value has not changed. In contrast, with SD active in the background, an ECU will know immediately when a service/another ECU is no longer providing a certain functionality. Failures are thus detected better and the respective failure modes can be activated within a certain timeframe.
- Additionally, SD can be used with individually **adjustable "Time-To-Live" (TTL) values**, which indicate how long an entry is valid. A user of that value expects an update once the TTL has expired. If it fails to arrive, the user can also conclude on the faulty behavior of the communication partner and can start specific error processing. This improves the stability of a network, but of course cannot replace messages in case cyclic data is missing for a safety critical application. Cyclic messages with "Application Cyclic Redundancy Checks" (CRCs) are normally used for an end-to-end safety application.
- **Dynamics in the case of partial networking for energy efficiency:** Because of the increasing size of the in-vehicle network and the increasing number of ECUs, energy efficiency is ever more important. As is pointed out in Section 6.3.3, it is of interest to fully power only those ECUs that are needed at a particular moment in time. This needs to take different scenarios into account. A customer might want to finish a call via the built-in hands-free system, despite having arrived at their destination and parked the car. The car should then be smart enough to deactivate all ECUs on the network that are not needed, like the engine control or drive train. This example shows that, with partial networking, the in-vehicle network can be expected to change dynamically. In the changing environment, active ECUs have to know which functions are still available and which are not. Without SD this can be realized with timeouts. Like in the start-up scenario, however, this makes the system slower than with SD. With SD the knowledge of availability is more immediate.

The more complex an in-vehicle network becomes, the better it is to have service-based communication and SD available. Without service-based communication, the complexity of an Automotive Ethernet network is much higher, as explained above. When implementing SD in a network, there are two principal approaches, a centralized and a decentralized one:

- In the **centralized approach**, one ECU monitors and maintains the service information of the network. Each participant sends its respective information just to this one ECU and each participant requests the respective information just from this one ECU.
- In the **decentralized approach** all communication partners apply the following rules: Each communication partner offers its available services to all other units via multicast or broadcast and each communication partner requests available services from all other units via multicast or broadcast. If two communication partners find each other, they can establish a respective one-to-one communication. Important advantages of the decentralized approach are: minimal start-up delays, which then mainly depend on the start-up time of the physical network; no specific third-party ECU is needed; and there are multiple sources of data, meaning that no single component has to

handle all the data but the load and risk in the event of an error are distributed. In other words, there exists no single point of failure.

Notes

1 The terms bridge, switch, and router are not always unambiguously used. This book shall use the terms similarly to what is being described in [100]. All terms – also including hubs and repeaters – refer to units in a communication network that pass on data. What differs is how and on what basis they do this.The use of the term **router** is comparably consistent. A router forwards data packets on the basis of ISO/OSI layer 3 addresses, which in today's networks are IP addresses. "Routing" describes a functionality. The term in itself does not say whether a router is a standalone box or a function integrated into a microcontroller. Generally, routers are used for large-scale communication: to pass on data between different Wide Area Networks (WANs), between different Local Area Networks (LANs), and also between LANs and WANs. In vehicles, routing is used in the flash and diagnostic automotive use case described in Section 3.1. In this use case, the central gateway holds the router that connects the car's internal "LAN," or so-called Automotive Area Network (AAN), to the outside world (see also Section 5.3 for the use of IP in an Ethernet-based communication in automotive). Routers can also be used to pass on data inside LANs, though this can generally be done more efficiently via what in this book shall be referred to as "switches."

The use of the term **switch** is not so consistent. In this book, a switch forwards packets in an Ethernet network based on the ISO/OSI layer 2 addresses, i.e., the hardware/MAC address provided in the Ethernet packet (see Section 1.2.1). A switch is directly related to the Ethernet technology and, at least in in-vehicle networking, a new concept. The impact of this concept on the in-vehicle EE architecture and topology choices is severe and subject of a separate chapter (Chapter 6). The switching function is generally realized in hardware in a special switch semiconductor. The semiconductor can be "switch only" (which is rare), a "switch with integrated PHYs" (which is most common) or a "switch integrated into a System on Chip" (SoC). If this book talks about switches, the default meaning is the (part of the) semiconductor that provides the respective function. In the IT industry, the term "switch" often refers to a standalone networking product, which actually has quite a market (see Table 1.3). This switch is a box with a number of RJ-45 sockets that allows the connection of various devices via Ethernet and that will direct the traffic between them (based on the layer 2 address). In Chapter 6, this book will also consider a separate ECU containing the switching function, which is then a "switch box" or "standalone switch." Sometimes the function of a switch is extended to layer 3 as a "layer 3 switch," which somewhat blurs the distinction and can be confusing. In this book, layer 3 forwarding is "routing" and layer 2 forwarding is "switching." Note that the IEEE 802.1 specifications never utilize the term "switch" as used here, but only "bridge."Following [101], a **bridge** also passes on traffic based on layer 2, but can do so not only at the MAC level between Ethernet links, but also on the Link Layer Control) level. This means, it can handle different MAC control algorithms and can thus bridge traffic between different IEEE technologies without needing the IP address. In the IEEE 802.1 terminology this distinction makes sense; for Automotive Ethernet it is of minor relevance. Occasionally the term "bridging" is also used on layer 3; after all, various different technologies can be bridged with the help of IP addressing. When using the term bridge in this book, it will always be used in combination with the layer it bases its functionality on – like "layer 2 bridge" – in order to avoid confusion.

Last, but not least, there are repeaters and hubs. Both function on layer 1, i.e., they have no intelligence that allows them to forward a packet based on addressing. A **repeater** is a simple one-to-one device that amplifies the signal to increase the range. Repeaters can be of interest in automotive, too, when, e.g., Ethernet PHYs designed for 10 or 15 m links are being used in trucks or busses. **Hubs**, in contrast, have lost their relevance in Ethernet networks in general, and will also not be considered in automotive. Hubs take the input data from one connected device and broadcast them to all attached other devices. This makes the link a shared link and contradicts the P2P/switched Ethernet network approach that has replaced CSMA/CD Ethernet a while back.

2 Since then Napster has been relaunched more than once. Latest, in June 2016, the music streaming service Rhapsody rebranded itself as Napster [102]. This shows that the memory of the revolutionary change Napster initiated in music consumption around the turn of the century, is still attributed with sufficient marketing potential 15 years later; even if user have to face the change from "for free" to "add-free" [100].

3 IEEE 802.1 Qat "Stream Reservation" as well as the IEEE 802.1 Qav "Traffic Shaping" were both incorporated in the IEEE 802.1Q revision of 2011 [37].

4 Universal Plug and Play (UPnP) describes a set of protocols that allow for the vendor-independent, distributed media management, discovery, and control in an IP-based network that consists of consumer devices like computers, printers, Internet gateways, audio rendering units, mobile devices, etc. The first version of UPnP was published as ISO/IEC 29341 in December 2008 [103] and updated/extended in 2011. The UPnP forum, which drives and markets the developments, was founded in October 1999. Since January 2016, the Open Connectivity Foundation (OCF) has taken over the assets from the UPnP Forum [104].

5 The Digital Living Network Alliance (DLNA), founded in June 2003, has the goal of ensuring the interoperability between applications of networked consumer devices that involve images, AV data [105]. For this, the DLNA provides design guidelines based on higher layer standards like UPnP and certification programs. The guidelines are based on standards like UPnP for media management, discovery, and control. At the time of writing the DLNA homepage listed about 1500 DLNA certified products.

6 Coordinated Shared Network (CSN) is a generic term for a network in which the media is shared on a contention-free, time-multiplexed basis. The network access in a CSN is coordinated by one unit designated or elected as the network coordinator, which also might be the interface to, e.g., an Ethernet LAN. CSN technologies are technologies used in the home environment – Multimedia over Coax (MoCa), Homeplug (Inhouse Powerline Communication), and Ultra Wide Band (UWB)/IEEE 802.15.4a are given as examples – and were thus considered necessary to integrate into the AVB concept [106] [34] [5]. CSNs play no role in automotive.

7 At the time, summer 2009, this was actually very progressive. Automotive Ethernet had not yet taken off. One car manufacturer, BMW, used Ethernet for flash updates and for a private link between HU and RSE (see Sections 3.1 and 3.2), but the interest of everyone else was very moderate. The bus system for automotive infotainment (AV) was MOST or the transmission was analog or LVDS, also at BMW. It can be assumed that the involvement of Harman and Broadcom in starting the AVnu Alliance spurred the inclusion of automotive. Harman had just provided the HU/RSE system for BMW and together with Broadcom BMW made promising progress toward the first use of 100 Mbps BroadR-Reach Ethernet.

8 The seven hops 2 ms requirement has two explanations. One is historical, the other was derived from the harshest, i.e., smallest, maximum network delay requirement in the professional auto domain. A musician needs to hear the response to his/her action within 10 ms; 8 ms of these are needed for DSP delays and the delay of the sound traveling, e.g., between speaker monitor and the musician. This leaves a delay of 2 ms for the network. These 2 ms were split into a realistic number of hops, leaving some margin and taking into consideration that in the worst case a 100 Mbps AVB packet needs to wait at every switch behind a best-effort 1500 bytes packet for about 122 μs [23]. Seven is also the historical limit for the layer 2 network hop count, which is reused in AVB. This seven-hop limit was established early on in Ethernet development. While IEEE 802.3 was working on repeaters, DEC developed

the first layer 2 bridge. The upper layer protocol DEC used for the bridge was time sensitive and did not support more than seven hops. A daisy chain of seven hops for a layer 2 bridged system seemed a reasonable worst case, and even though the DEC upper layer protocol is no longer used and technology has advanced to fast hardware supported switching, the number has stayed [6].

9 Reference [31] includes the IEC 61883-2,4,6,7,8 formats (i.e., Standard Definition Digital Video Cassette Recorder (SD-DVCR), MPEG2-Transport Stream of compressed video, uncompressed digital audio and music, satellite TV MPEG ITU-R BO.1294, and digital video data ITU-R BT.601), IIDC 1394-based uncompressed industrial camera, and formats that will be defined by the Musical Instrument Digital Interface (MIDI) manufacturers' association [107].

10 The 2016 IEEE 1722 version includes a Clock Reference Format (CRF), which allows for the distribution of event timing information within a system. This is of particular interest for Driver ASsist (DAS) functions, e.g., to be able to combine four camera pictures correctly in a surround view image. It would also allow to be able to determine the exact distance to an obstacle on the road from camera images. For autonomous driving cameras represent an important source of redundant information in addition to other sensors like radar, ultrasound, etc.

11 It sounds unusual, but especially in automotive this can happen easily. When, e.g., a car is parked in a garage where it has no GPS reception, the clock of the navigation unit might be inferior to the clock of another ECU. When the car then leaves the garage and GPS becomes available, the clock of the navigation system might suddenly be superior.

12 The Multiple VLAN Registration Protocol (MVRP) provides for the registration into the correct VLAN(s) and Multiple MAC Registration Protocol (MMRP) provides for the registration and announcement of the multicast addresses.

13 Note that when writing the first edition of this book, it was anticipated that Class C traffic would be specified to have a packet frequency of 1 kHz. This has not materialized. It was seen as more fitting to provide a traffic class based on a number of audio samples well aligned with processing capabilities, which would then have a varying packet frequency, depending on whether it was used with 44.1 kHz audio sampling or 48 kHz [38].

14 In the automotive industry, FlexRay was developed exactly to fulfill the requirements of safety critical applications (see also Section 2.2.5.2) and it might well still be used for that purpose. However, this book is about Ethernet-based communication and thus discusses the possibilities an Ethernet system provides in the same context.

15 As was introduced in Section 1.2.2, the synchronization accuracy in industrial applications is in the range of 1 μs [108]. Obviously, if a 12.2 μs packet needs to complete the transmission first, there is no chance to meet the requirement and use Ethernet for this application.

16 IEEE 802.3 limits the Ethernet frame size to 1500 bytes payload. However, a concept called "jumbo frames" exists, which is used in various nonstandardized variations and allows payloads of up to 9000 bytes payload [109]. These packets intend to increase throughput by reducing overhead. The potential delay owing to packets blocking the egress port, however, is increased almost 6-fold. Today, this is one of the reasons not to use jumbo frames. Once IET and preemption found their way into products, the authors expect that also jumbo frames will be used more frequently, also in automotive.

17 IEEE 802.1 standardized a number of security related protocols, whose reuse for Automotive Ethernet is worth investigating, e.g., IEEE 802.1x is very widely implemented for key management [74]. It defines Ethernet encapsulation for the Extensible Authentication Protocol (EAP), which in return is a framework for the exchange of authentication messages. Another standard of interest is IEEE 802.1AR, the Secure Device Identity Standard, which was first published in 2009 and updated in 2015. It defines the device identity and cryptography to be used by the device and the operation within EAP-TLS/802.1x. It assumes hardware support for efficient operation.

www.cargeek.ir

18 Other protocols of the TCP/IP protocol suite are the Address Resolution Protocol (ARP) and the Reverse Address Resolution Protocol (RARP), which translate layer 3 IP addresses to layer 2 Ethernet MAC addresses and vice versa (RFC 826, 1982 [110]). Furthermore, the Internet Control Message Protocol (ICMP) sends error messages or relays query messages (RFC 792, 1981 [111]). The Internet Group Management Protocol (IGMP) establishes IP multicast group membership for IPv4 (RFC 3376, 2002 [112]) on layer 2 MAC addresses. When using static multicast addresses, it is not necessary to provide IGMP for an Automotive Ethernet application. Last, but not least, the User Datagram Protocol (UDP) (RFC 768, 1980 [87]) is part of the TCP/IP protocol suite. See also Figure 5.1.

19 Subnetting allows the division of a single Class A, B, or C network into smaller networks. After all, a Class A network with up to 16.78 Mio units in one network, or even 65.534 in a Class B network, is a lot of units. Means to simplify routing and network design are thus appealing. Subnetting was first specified 1985 in RFC 950 [113]. Variable Length Subnet Mask (VLSM) allows a subnetted network to use more than one subnet mask and thus to use the assigned address space more efficiently. It was first addressed 1987 in RFC 1009 [114]. Classless Inter-Domain Routing (CIDR) eliminated the concept of Class A, B, and C addresses. With CIDR, the number of nodes in a network was no longer restricted to either 16.78 Mio (Class A), 65.534 (Class B), or 254 (Class C), but was selectable arbitrarily. CIDR also allowed the reduction of the number of entries in routing tables. It is said that, without this, the Internet would not have sustained [114]. CIDR was specified in 1993 RFC 1517, 1518, 1519, and 1520 [89]. With IPv6, neither VLSM nor CIDR are needed.

20 The service *CD_Player* is used as an example to explain the basic features of SOME/IP and RPCs. Every service has to be defined during the development process by its service interface. This is normally done with an Interface Description Language (IDL) and could look as follows:

```
Service CD_Player
{
track_number          // Field
{unsigned int track;  // the track number
set (track);          // Method for setting the track (uses a request/response method)()
get ();               // Method for getting the actual track number played
}
tray.eject ();        // Event that is triggered if the eject button is pressed
Boolean tray_state;   // Status OPEN or CLOSED when tray is open or closed
                         respectively
tray_state: open_tray // Method that is used for open the tray, the return value of this
   ();                   Method is
// the tray_state.
}
```

Following the above service interface definition, "A client would like to change the track to track number 10" would cause the command *CD_Player.track_number.set(10)* to be sent from, e.g., the Head Unit (HU, client) to the CD-player (server). The method of the service is *track_number.set*, the payload value is *10*, and the communication principle typically used for this would be a request/response method, meaning that a response for the command set is expected.

In the next case "A client would like to open the tray of the CD-Player and would like to know when the job is done." When using the above description, the command from any client (e.g., the Head Unit) to the CD-player would be *CD_Player.open_tray()*. In this example the client expects a response in the form of an acknowledgment. This is thus an example of the request & response communication principle. When the client receives *CD_Player.open_tray() = = OPEN*, it knows that the command has been successfully completed.

There is more than one way to achieve the same result. The key, whichever way is chosen, in service-based communication was clearly defined upfront with respective data types, the data structures, and the methods and communication principles used. In the above example the client could also send a read ("get field") command to receive information on the CD-player tray status. Or it could have subscribed to the CD-player, asking the CD-player to automatically inform the client every time the status of the tray changes ("event"). The respective command would be *Subscribe.CD_Player.Eject()*. In the event of the tray opening the CD-player would send *CD_Player.Eject()* to all subscribed clients, which then would be a event from the server to the clients.

This example emphasizes the possibilities service-based communication offers in contrast to the CAN-like communication principle of fire & forget and simple messages only. The serialization of SOME/IP ensures that the information fits into the existing packet format like all other traffic. The content, however, ensures a type of contract for a service to be fulfilled between the communication parties.

References

[1] L. Völker, "One for All: Interoperability from AUTOSAR to GENIVI," in *1st Ethernet&IP@Automotive Technology Day*, Munich, 2011.

[2] ITWissen, "QoS (Quality of Service)," continuously updated. [Online]. Available: www.itwissen.info/definition/lexikon/quality-of-service-QoS-Dienstguete.html. [Accessed 15 August 2016].

[3] ITU, "Definitions of Terms Related to Quality of Service," ITU, Geneva, 2008.

[4] ITU-T, "Network Performance Objectives for IP-Based Services," ITU, Geneva, 2011.

[5] IEEE 802.3, "Residential Ethernet, IEEE 802.3 Call for Interest," July 2004. [Online]. Available: http://grouper.ieee.org/groups/802/3/re_study/public/200407/cfi_0704_1.pdf. [Accessed 12 October 2013].

[6] Y. Kim, *E-mail correspondence*, 2013.

[7] T. Lamont, "Napster: The Day the Music Was Set Free," 24 February 2013. [Online]. Available: www.theguardian.com/music/2013/feb/24/napster-music-free-file-sharing. [Accessed 30 October 2013].

[8] Gartner, "Gartner Says Declining Worldwide PC Shipments in Fourth Quarter of 2012 Signal Structural Shift of PC Market," 14 January 2013. [Online]. Available: www.gartner.com/newsroom/id/2301715. [Accessed 9 September 2013].

[9] M. Johas Teener, "No-excuses Audio/Video Networking: the Technology Behind AVnu," 24 August 2009. [Online]. Available: http://avnu.org/wp-content/uploads/2014/05/No-excuses-Audio-Video-Networking-v2.pdf. [Accessed 20 August 2016].

[10] IEEE 802.3, "Website for IEEE 802.3 Residential Ethernet Study Group," 21 November 2005. [Online]. Available: http://grouper.ieee.org/groups/802/3/re_study/. [Accessed 11 October 2013].

[11] R. Kreifeld, *E-mail correspondence*, 2013.

[12] Business Wire, "AVnu Alliance Launches to Advance Quality of Experience for Networked Audio and Video," 25 August 2009. [Online]. Available: www.businesswire.com/news/google/20090825005929/en. [Accessed 8 October 2013].

[13] Marvell, "Marvell Announces Industry-First Audio Video Bridging Family of SoCs with Integrated Switching, CPU and Endpoint Functionality," 8 May 2012. [Online]. Available: www.marvell.com/company/news/pressDetail.do?releaseID=2296. [Accessed 16 August 2016].

[14] J. Urban, "Debunking Some Myths about AVB," 17 May 2013. [Online]. Available: http://blog.biamp.com/debunking-some-myths-about-avb/. [Accessed 20 August 2016].

[15] IEEE 802.1, "802.1 Plenary – 11/2012 San Antonio Closing Slides," November 2012. [Online]. Available: www.ieee802.org/1/files/public/minutes/2012–11-closing-plenary-slides.pdf. [Accessed 30 October 2013].

[16] E. Hellerud, "Transmission of High Quality Audio over IP Networks," April 2009. [Online]. Available: www.diva-portal.org/smash/get/diva2:277816/FULLTEXT02. [Accessed 1 November 2013].

[17] R. Kreifeld, "AVB for Professional A/V Use," 30 July 2009. [Online]. Available: http://avnu.org/wp-content/uploads/2014/05/AVnu-Pro__White-Paper.pdf. [Accessed 20 October 2013].

[18] H. Kaltheuner, "Das Universalnetz, Ethernet AVB: Echtzeitfähig und Streaming-tauglich," *c't*, no. 13, pp. 176–81, June 2013.

[19] M. Johas Teener, "Residential Ethernet Objectives, Requirements and Possible Solutions," 9 May 2009. [Online]. Available: www.ieee802.org/1/files/public/docs2005/liaison-mikejt-rese-objectives-requirements-0505.pdf. [Accessed 17 October 2013].

[20] R. Steinmetz, "Human Perception of Jitter and Media Synchronization," *IEEE Journal on Selected Areas in Communication*, vol. 14, no. 1, pp. 61–72, January 1996.

[21] K. Stanton, "AVB for Home/Consumer Electronics Use," 11 August 2009. [Online]. Available: http://avnu.org/wp-content/uploads/2014/05/AVB-for-Home-Consumer-Electronics-Use__White-Paper.pdf. [Accessed 24 October 2013].

[22] F. Held, "Digitale Audio-, Video- & Licht-Kommunikationsprotokolle in der Veranstaltungstechnik," 12 October 2012. [Online]. Available: https://entropia.de/wiki/images/b/b8/Avb_vortrag_felix.pdf. [Accessed 24 October 2013].

[23] M. Johas Teener, *E-mail correspondence*, 2013.

[24] R. Boatright, "Understanding New Audio Video Bridging Standards," 10 May 2009. [Online]. Available: www.embedded.com/design/connectivity/4027005/Understanding-IEEE-s-new-audio-video-bridging-standards-item-1. [Accessed 24 October 2013].

[25] M. Schettke, "AVB Audionetzwerke in der Praxis: Betriebssicherheit und deren automatisierte Überprüfung," 10 June 2014. [Online]. Available: http://schettke.com/files/Diplomarbeit_AVB-Praxis_Schettke_2014.pdf. [Accessed 10 August 2016].

[26] R. Kreifeld, "AVB for Automotive Use," 20 July 2009. [Online]. Available: only updated version available http://avnu.org/wp-content/uploads/2014/05/2014–11–20_AVnu-Automotive-White-Paper_Final_Approved.pdf. [Accessed 20 October 2013].

[27] M. Kicherer and T. Königseder, "BMW Proposal for an AVB Gen 2 Automotive Profile," BMW White Paper, München, 2013.

[28] K. Matheus, M. Kicherer, and T. Königseder, "Audio/Video Transmission in Cars using Ethernet," BMW White Paper, Munich, 2010.

[29] T. Hogenmüller et al., "Use Cases & Requirements for IEEE 802.3 RTPGE Ethernet," May 2012. [Online]. Available: http://grouper.ieee.org/groups/802/3/RTPGE/public/may12/hogenmuller_01_0512.pdf. [Accessed 30 May 2012].

[30] K. Matheus et al., "1 Pair or 2 Pairs for RTPGE: Impact on System Other Than the PHY Part 1: Weight & Space," January 2013. [Online]. Available: www.ieee802.org/3/bp/public/jan13/matheus_3bp_01_0113.pdf. [Accessed 18 January 2013].

[31] IEEE Computer Society, "IEEE 1722–2016: Transport Protocol for Time-Sensitive Applications in Bridged Local Area Networks," IEEE, New York, 2016.
[32] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," July 2003. [Online]. Available: http://tools.ietf.org/html/rfc3550. [Accessed 15 November 2013].
[33] IEEE Computer Society, "IEEE 802.1AS-2011: Timing and Synchronization for Time-Sensitive Applications in Bridged Local Area Networks," IEEE, New York, 2011.
[34] IEEE Computer Society, "IEEE 1588–2008: Precision Clock Synchronization Protocol for Networked Measurement and Control Systems," IEEE, New York, 2008.
[35] Hirschmann, "Precision Clock Synchronization, the Standard IEEE 1588," 2008. [Online]. Available: www.belden.com/docs/upload/precision_clock_synchronization_wp.pdf. [Accessed 28 October 2013].
[36] EndRun Technologies, "Precision Time Protocol (PTP/IEEE-1588)," 2013. [Online]. Available: www.endruntechnologies.com/pdf/PTP-1588.pdf. [Accessed 8 November 2013].
[37] IEEE Computer Society, "IEEE 802.1Q-2011: Media Access Control (MAC) Bridges and Virtual Bridge Local Area Networks," IEEE, New York, 2011.
[38] G. Bechtel, M. K. Ben Gale, and D. Olsen, "Automotive Ethernet AVB Functional and Interoperability Specification, Revision 1.4," AVnu, Beaverton, 2015.
[39] IEEE Computer Society, "IEEE 802.1Qav:2009 Amendment 12: Forwarding and Queuing Enhancements for Time-Sensitive Streams," IEEE, New York, 2009.
[40] S. Stein and R. Racu, "Ethernet Quality of Service @ Volkswagen," in *2nd Ethernet&IP@Automotive Technology Day*, Regensburg, 2012.
[41] J. Diemer, D. Thiele, and R. Ernst, "Formal Worst-Case Timing Analysis of Ethernet Topologies with Strict-Priority and AVB Switching," in *7th IEEE International Symposium on Industrial Embedded Systems (SIES12)*, Karlsruhe, 2012.
[42] IEEE Computer Society, "IEEE 1733–2011: Layer 3 Transport Protocol for Time-Sensitive Applications in Local Area Networks," IEEE, New York, 2011.
[43] J. Damori, "Are Layer 2 or Layer 3 Protocols Better? Yes," 31 May 2013. [Online]. Available: http://blog.biamp.com/are-layer-2-or-layer-3-protocols-better-yes/. [Accessed 20 October 2013].
[44] IEEE Computer Society, "IEEE 802.1BA-2011: Audio Video Bridging (AVB) Systems," IEEE, New York, 2011.
[45] IEEE Computer Society, "IEEE 1722.1–2013: Device Discovery, Connection Management, and Control Protocol for IEEE 1722 Based Devices," IEEE, New York, 2013.
[46] J. Lane, "Digital Audio Listening In Car Is Increasing," 11 October 2010. [Online]. Available: http://audio4cast.com/2010/10/11/digital-audio-listening-in-car-is-increasing/. [Accessed 12 November 2013].
[47] M. Jochim and M. Osella, "The Need for IEEE Standardized Ethernet Mechanisms for Active Safety Applications," in *3rd Ethernet&IP@Automotive Technology Day*, Leinfelden-Echterdingen, 2013.
[48] D. Zebralla, "Requirements on Future in-Vehicle Architectures for Automotive Ethernet," in *Automotive Ethernet Congress*, München, 2016.
[49] M. Johas Teener, "IEEE 802 Time-Sensitive Networking: Extending Beyond AVB," in *3rd Ethernet & IP @ Automotive Technology Day*, Leinfelden-Echterdingen, 2013.
[50] D. Pannell, "IEEE TSN Standards Overview & Update," in *5th Ethernet&IP@Automotive Technology Day*, Yokohama, 2015.

[51] IEEE-SA, "IEEE 802.3br-2016 Amendment: Specification and Management Parameters for Interspersing Express Traffic," IEEE, New York, 2016.

[52] IEEE Computer Society, "IEEE 802.1Qbu: Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks – Amendment: Frame Preemption," IEEE, New York, 2016.

[53] Wikipedia, "TTEthernet," 31 March 2016. [Online]. Available: http://en.wikipedia.org/wiki/TTEthernet. [Accessed 20 August 2016].

[54] IEEE Computer Society, "IEEE 802.1Qbv: Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks Amendment: Enhancements for Scheduled Traffic," IEEE, New York, 2016.

[55] IEEE 802.1, "Website for .1Qch – Cyclic Queuing and Forwarding," continuously updated. [Online]. Available: www.ieee802.org/1/pages/802.1ch.html. [Accessed 20 August 2016].

[56] M. Jochim, "Ingress Policing," 10–14 November 2013. [Online]. Available: www.ieee802.org/1/files/public/docs2013/tsn-jochim-ingress-policing-1113-v2.pdf. [Accessed 20 August 2016].

[57] IEEE 802.1, "Website for: 802.1Qci – Per-Stream Filtering and Policing," continuously updated. [Online]. Available: www.ieee802.org/1/pages/802.1ci.html. [Accessed 3 September 2016].

[58] IEEE Computer Society, "IEEE 802.1Qca: Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks Amendment: Path Control and Reservation," IEEE, New York, 2016.

[59] IEEE 802.1, "Website for 802.1CB – Frame Replication and Elimination for Reliability," continuously updated. [Online]. Available: www.ieee802.org/1/pages/802.1cb.html. [Accessed 20 August 2016].

[60] IEEE 802.1, "Website for 802.1AS-Rev – Timing and Synchronization for Time-Sensitive Applications," continuously updated. [Online]. Available: www.ieee802.org/1/pages/802.1AS-rev.html. [Accessed 20 August 2016].

[61] IEEE 802.1, "Website for 802.1Qcc – Stream Reservation Protocol (SRP) Enhancements and Performance Improvements," continuously updated. [Online]. Available: www.ieee802.org/1/pages/802.1cc.html. [Accessed 20 August 2016].

[62] R. Moussalli, "Dealing with security threats in the digital age: Part 1," 19 August 2015. [Online]. Available: http://tatacommunications-newworld.com/2015/08/dealing-with-security-threats-in-the-digital-age-part-1/. [Accessed 29 August 2016].

[63] C. Miller and C. Valasek, "Remote Exploitation of an Unaltered Passenger Vehicle," 10 August 2015. [Online]. Available: http://illmatics.com/Remote%20Car%20Hacking.pdf. [Accessed 5 October 2016].

[64] A. Greenberg, "Remote Exploitation of an Unaltered Passenger Vehicle," 21 July 2015. [Online]. Available: www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/. [Accessed 5 October 2016].

[65] SANS Institute, "The Best Security Books to have in your library," SANS Institute, 2014. [Online]. Available: www.sans.edu/research/book-reviews/article/security-books-best. [Accessed 2 February 2014].

[66] S. Singer, "IP based Communication in Vehicles – learnings from the IT Industry," in *Automotive Ethernet Congress*, Munich, 2016.

[67] AUTOSAR, "Specification of Module Secure Onboard Communication, Release 4.2.2," undated. [Online]. Available: www.autosar.org/fileadmin/files/releases/4-2/software-architecture/safety-and-security/standard/AUTOSAR_SWS_SecureOnboardCommunication.pdf. [Accessed 29 August 2016].

[68] R. Pallierer and M. Ziehensack, "Secure Ethernet for Autonomous Driving," in *Automotive Ethernet Congress*, München, 2016.

[69] Wikipedia, "Transport Layer Security," 29 August 2016. [Online]. Available: https://en.wikipedia.org/wiki/Transport_Layer_Security. [Accessed 30 August 2016].

[70] T. Dierks and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2," August 2008. [Online]. Available: https://tools.ietf.org/html/rfc5246. [Accessed 30 August 2016].

[71] M. Lindner, "Security Architecture for IPsec," 2007. [Online]. Available: www.ict.tuwien.ac.at/lva/384.081/infobase/L97-IPsec_v4-7.pdf. [Accessed 2 February 2014].

[72] Wikipedia, "IPsec," 20 July 2016. [Online]. Available: http://en.wikipedia.org/wiki/Ipsec. [Accessed 20 August 2016].

[73] IEEE Computer Society, "IEEE 802.1AEbw-2013: Media Access Control (MAC) Security Amendment 2: Extended Packet Numbering," IEEE, New York, 2013.

[74] Y. Kim, "Ethernet Security," in *Automotive Ethernet Congress*, Munich, 2016.

[75] Wikipedia, "Books on Cryptography," 18 August 2016. [Online]. Available: https://en.wikipedia.org/wiki/Books_on_cryptography. [Accessed 31 August 2016].

[76] L. Apvrille, R. El Khayari, O. Henniger, Y. Roudier, H. Schweppe, H. Seudié, B. Weyl, and M. Wolf, "Secure Automotive on-Board Electronics Network Architecture," 30 May 2010. [Online]. Available: www.evita-project.org/Publications/AEHR10.pdf. [Accessed 8 October 2016].

[77] H. G. Molter, "Introduction to Security," in *Automotive Ethernet Congress*, Munich, 2016.

[78] SAE, "Cybersecurity Guidebook for Cyber-Physical Vehicle Systems, J3061," continuously updated. [Online]. Available: http://standards.sae.org/wip/j3061/. [Accessed 2 September 2016].

[79] ISO, "ISO/AWI 21434 Road Vehicles – Automotive Security Engineering," 2016. [Online]. Available: www.iso.org/iso/catalogue_detail.htm?csnumber=70918&utm_source=ISO&utm_medium=RSS&utm_campaign=Catalogue. [Accessed 18 November 2016].

[80] H. Goto, "In-Vehicle Ethernet Technology Promoted by JASPAR and Industry Trends," in *4th Nikkei Electronics/Automotive Seminar*, Tokyo, 2016.

[81] B. Gale, "Ethernet Security in the Car," in *IEEE-SA (4th) Ethernet&IP@Automotive Technology Day*, Detroit, 2014.

[82] D. Kleidermacher and M. Kleidermacher, *Embedded Systems Security – Practical Methods for Safe and Secure Software and Systems Development*, Oxford: Newnes, 2012.

[83] W. R. Stevens, *TCP/IP Illustrated*, vol. 1, *The Protocols*, Reading, MA: Addison Wesley Longman, 1994.

[84] Wikipedia, "Internet Protocol Suite," 19 August 2016. [Online]. Available: http://en.wikipedia.org/wiki/Internet_protocol_suite. [Accessed 20 August 2016].

[85] Information Sciences Institute University of Southern California, "Transmission Control Protocol," September 1981. [Online]. Available: http://tools.ietf.org/html/rfc793. [Accessed 4 August 2013].

[86] Information Sciences Institute University of Southern California, "Internet Protocol," September 1981. [Online]. Available: http://tools.ietf.org/html/rfc791. [Accessed 4 August 2013].

[87] J. Postel, "User Datagram Protocol," 29 August 1980. [Online]. Available: http://tools.ietf.org/html/rfc768. [Accessed 4 August 2013].

[88] M. Kessler, "Ethernet in Small ECUs, Challenges and Chances," in *3rd Ethernet&IP@Automotive Technology Day*, Leinfelden-Echterdingen, 2013.

[89] 3Com, "Understanding IP Addressing, Everything You Ever Wanted to Know," 2001. [Online]. Available: http://pages.di.unipi.it/ricci/501302.pdf. [Accessed 20 August 2016].

[90] Number Resource Organization, "Free Pool of IPv4 Address Space Depleted," 3 February 2011. [Online]. Available: www.nro.net/news/ipv4-free-pool-depleted. [Accessed 23 November 2013].

[91] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6)," December 1995. [Online]. Available: http://tools.ietf.org/html/rfc1883. [Accessed 23 November 2013].

[92] C. Bao, C. Huitema, M. Bagnulo, M. Boucadair, and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators," October 2010. [Online]. Available: http://tools.ietf.org/html/rfc6052. [Accessed 18 February 2014].

[93] A. Busnelli, "Car Software: 100M Lines of Code and Counting," 26 June 2014. [Online]. Available: www.linkedin.com/pulse/20140626152045–3625632-car-software-100m-lines-of-code-and-counting. [Accessed 18 August 2016].

[94] The Apache Software Foundation, "Apache Etch," 2013. [Online]. Available: http://etch.apache.org/. [Accessed 19 February 2014].

[95] A. Bouard, J. Schanda, D. Herrscher, and C. Eckert, "Automotive Proxy-Based Security Architecture for CE Device Integration," in *Mobile Wireless Middleware, Operating Systems, and Applications*, Heidelberg: Springer, 2013, pp. 62–76.

[96] K. Weckemann, F. Satzger, L. Stolz, D. Herrscher, and C. Linnhoff-Popien, "Lessons from a Minimal Middleware for IP-Based in-Car Communication," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, Alcala de Henares, 2012.

[97] Google Developers, "Protocol Buffers," 2 April 2012. [Online]. Available: https://developers.google.com/protocol-buffers/. [Accessed 30 January 2014].

[98] Apple, "Bonjour for Developers," 2014. [Online]. Available: https://developer.apple.com/bonjour/. [Accessed 30 January 2014].

[99] L. Völker, "Scalable service-Oriented Middleware over IP (SOME/IP)," 2013. [Online]. Available: http://some-ip.com. [Accessed 30 January 2014].

[100] Napster, "Company Info," continuously updated. [Online]. Available: http://us.napster.com//about?from=rhapsody. [Accessed 15 August 2016].

[101] T. Hümmler, "Router, Switches, Hub und Co. Ratgeber: Was ist im Netzwerk?," 26 October 2013. [Online]. Available: www.tecchannel.de/netzwerk/lan/2038788/router_repeater_switch_hub_bridge_was_ist_was_im_netzwerk/. [Accessed 18 October 2013].

[102] B. Popper, "Rhapsody Rebrands Itself as Napster because ¯_(ツ)_/¯," *The Verge*, 14 June 2016. [Online]. Available: www.theverge.com/2016/6/14/11936974/rhapsody-rebrands-as-napster. [Accessed 15 August 2016].

[103] ISO/IEC, "ISO/IEC 29341- (1–13)-1:2008: Information Technology – UPnP Device Architecture," ISO, Geneva, 2008.

[104] Open Connectivity Forum (OCF), "About UPnP," 1 January 2016. [Online]. Available: https://openconnectivity.org/upnp. [Accessed 23 August 2016].

[105] DLNA, "About DLNA," 2013. [Online]. Available: www.dlna.org/dlna-for-industry/about-dlna. [Accessed 1 November 2013].

[106] P. Klein, "Support for Coordinated Shared Network in 802.1AVB," January 2008. [Online]. Available: www.ieee802.org/1/files/public/docs2008/av-phkl-csn-0108-v1.pdf. [Accessed 1 November 2013].

[107] Musical Instrument Digital Interface Manufacturers Association (MIDI), "IEEE Ethernet AVB, AVB and MIDI," 2013. [Online]. Available: www.midi.org/techspecs/avbtp.php. [Accessed 2 November 2013, no longer available].

[108] M. Felser and T. Sauter, "Standardization of Industrial Ethernet – the Next Battlefield?," in *Proceedings of the IEEE 5th International Workshop on Factory Communication Systems*, Vienna 22 September 2004.

[109] Wikipedia, "Jumbo frame," 16 September 2016. [Online]. Available: https://en.wikipedia.org/wiki/Jumbo_frame. [Accessed 5 October 2016].

[110] D. C. Plummer, "An Ethernet Address Resolution Protocol," November 1982. [Online]. Available: http://tools.ietf.org/html/rfc826. [Accessed 21 November 2013].

[111] J. Postel, "Internet Control Message Protocol," September 1981. [Online]. Available: http://tools.ietf.org/html/rfc792. [Accessed 21 November 2013].

[112] B. Cain, S. Deering, I. Kouvelas, B. Fenner, and A. Thyagarajan, "Internet Group Management Protocol, Version 3," October 2002. [Online]. Available: http://tools.ietf.org/html/rfc3376. [Accessed 21 November 2013].

[113] J. Mogul and J. Postel, "Internet Standard Subnetting Procedure," August 1985. [Online]. Available: www.ietf.org/rfc/rfc950.txt. [Accessed 23 November 2013].

[114] R. Braden and J. Postel, "Requirements for Internet Gateways," June 1987. [Online]. Available: www.ietf.org/rfc/rfc1009.txt. [Accessed 23 November 2013].

www.cargeek.ir

6 Ethernet in Automotive System Development

The decision on the overall functionality to provide with a car is first of all a marketing decision and independent of the capabilities of in-vehicle networking technologies used and available. However, as soon as decisions have to be made on how to enable the functionality, in-vehicle networking becomes important. Aspects like flexibility, scalability, or how to distribute functions are severely impacted by the properties of in-vehicle networking. This chapter thus discusses the opportunities and changes Automotive Ethernet brings to system development.

6.1 A Brief Overview of the System Development Process

The development process in automotive follows the V-cycle. While Section 2.3.1 used the V-cycle to explain the responsibilities shared between car manufacturer and Tier 1 supplier, in this section the model is used to explain the changes the introduction of Ethernet-based communication brings to the development process. The important idea of the V-cycle is to follow a top-down approach on the development side and a bottom-up approach on the test side. On both sides, each new step requires the conclusion of the previous. During the development, the later need for testing is directly supported with the provision of test cases. This ensures stringent test coverage.[1]

In the context of in-vehicle networking, it is not necessary to consider the complete car development process. Instead a focus on the development of the Electric and Electronics (EE or E/E) architecture is sufficient (see also [1]). The task of an automotive EE architecture is to enable all required (electric and electronic) functions in a vehicle, while fulfilling the constraints given by cost targets and space limitations. Figure 6.1 gives an overview of the respective elements of the V-cycle. In the following, each step is briefly described, before the next sections concentrate on those steps affected by the introduction of Automotive Ethernet.

In the first step, product management and sales define the (EE enabled) system requirements, i.e., the customer functions a new car model should have. This includes the distinction between functions installed in every car and functions that can be bought by customers as options. Directly or indirectly, this definition also includes the interdependencies that might exist between some of the functions. In a complementary case, a customer might only be able to buy a certain function if he or she bought another function at the same time. An example is a rear view camera that can only be bought if

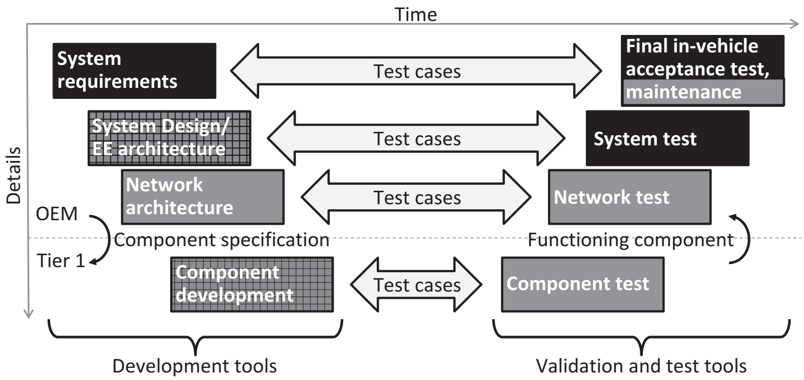

Figure 6.1 Overview of the elements of the automotive EE development V-cycle. The gray boxes indicate the areas with most changes; the checkered boxes indicate the areas in which there are some Ethernet-induced changes concerning the software.

the customer also selects a Head Unit (HU) with a suitable display. In an exclusive case, a customer cannot select two options simultaneously, e.g., the rear view camera cannot be bought, if the customer wants a HU without appropriate display. There are various reasons to provide specific functions in a car. The most obvious is to offer functions the customers want to have. Others are profitability, image, legislative requirements, or the wish to achieve a certain safety rating by, e.g., the European New Car Assessment Program (Euro NCAP) [2]. This first step of the development process does not define how the functions are being enabled, but represents the ideal result of the system [3]. With the next steps that look into the feasibility, it might turn out that this ideal result is not achievable. The system is then adapted accordingly.

The second step defines the EE architecture. In general, this means that the system engineers propose a solution on how the requirements can be implemented. First, all customer functions are broken down into smaller functional entities, so-called “function blocks.” Each function block describes one part of a customer function at a certain level. For the surround view system, for example, one function block starts the surround view function (e.g., the use of the reverse gear), one each records the images to front, sides, and back, one combines the images, one identifies and marks pedestrians in those images, and, finally yet importantly, one displays the image to the driver.

Once all function blocks have been defined, the very crucial and complex task of partitioning the function blocks onto ECUs, sensors, and actuators follows. In an ideal world, this step is a cost optimization process, in which function blocks are clustered onto an optimal number of ECUs, sensors, and actuators depending on the basic/optional definition given by the requirements of step one. Input values to this process would be expected customer take rates (potentially including roll out plans over different car models), functional safety targets (ASIL levels, see ISO 26262 [4]), and preferences toward a more integrated or a more distributed approach (see Section 6.3.2.1). The next steps would then define how the ECUs, sensors, and actuators are powered and communicate with each other and what the 3D routing layout, as described below, would look like.

In a not so ideal world, however, there are lots of interdependencies with the following as well as the preceding steps. If, because of the spatial constraints of a certain car model, an optional function can be offered only when integrated into an ECU with basic functions, this basic ECU might become too expensive to meet the cost targets. Or, the proposed partitioning might stress the available data rate on the preferred networking technology and the targeted communication technology cannot be used. This in return affects the costs and the price target might not be met. Or, the function is too new to have a sound cost analysis and later developments show that changes are necessary to the price target. In the real world, partitioning functions onto ECUs is thus an iterative process in which the EE architecture is defined via various feedback loops. The output of this step is a description of ECUs, sensors, and actuators, as well as their interdependencies and communication requirements.[2]

Breaking up the customer functions and partitioning them onto ECUs as such is not affected by the introduction of Automotive Ethernet. Nevertheless, in combination with the selection of the networking technology, Automotive Ethernet might well impact the system design: The higher data rate might allow for a different partitioning of functions, e.g., onto fewer ECUs with more data exchange between them.

The EE architecture directly affects the network architecture. The network architecture defines the in-vehicle communication and the power supply. The in-vehicle communication describes which ECU is connected via which communication/IVN technology to the rest of the system. Also, the intertechnology communication via gateways and their location in the network is defined. Supported data and data rates, timing behavior, and quality are important criteria for the selection of the in-vehicle networking technology. Naturally, the introduction of Ethernet causes quite some changes to the in-vehicle communication. Specific implications are thus discussed in more detail in Section 6.3.2. The power supply architecture defines how all units receive stable power in an energy-efficient manner. Section 6.3.3 discusses the changes that Ethernet induces here.

The network architecture has two further outputs. One is the spatial layout of the network architecture, i.e., the 3D position of the ECUs, sensors, and actuators, the routes for the harness elements that connect them, and the terminations for the in-vehicle networking technologies. Important considerations for the spatial layout are weight, installation space, harness diameter, maximum link lengths, inline connections between different areas of the car, pinning rules, whether the unit is in dry or wet areas, in high or very high temperature areas, and diagnosability. After all, in the event of malfunctions, all units and elements of the harness need to be accessible in a garage. The changes Automotive Ethernet induces on connectors, wiring, or harness manufacturing have been addressed in Section 4.2.

The second output is that the definition of the ECUs provided with the EE architecture also gives indications on the communication between function blocks within an ECU and thus of the design of the latter. While today the final design is done by the Tier 1 supplier of the ECU, car manufacturers can use the information for a first cost estimate and to provide some guidelines to the supplier.

Based on the provided information, the supplier then completes the design of the ECU and implements it. This includes hardware as well as software design, which is indeed impacted by Ethernet (see Section 6.2). The development of each ECU follows its own V-cycle. To use Ethernet communication within an ECU is also possible: It is a design choice that can be made. However, it has to be integrated like any other networking function. In the case of high data rates, generally Direct Memory Access (DMA) mechanisms need to be included for hardware acceleration. Intelligent approaches enable this without the use of special hardware. This is not limited to Ethernet, however, but is the same for other busses like I^2C and therefore the topic does not require special attention in this chapter.

On the test side, first the ECUs as such are tested. Next, when connecting the ECUs, (only) the networking functions like start-up and shutdown are tested. Only when these have been successful, the combined user functionalities are tested, before the final tests are performed inside the car. Last, but not least, the car and its network need to be maintained throughout the lifetime of the car. The impact Automotive Ethernet has on the component test, the network test, and the maintenance are described in Section 6.4. The system-level tests as well as the in-vehicle tests are not really affected by the introduction of Automotive Ethernet.

6.2 The Software Design

Two different software design approaches currently prevail in the automotive industry, AUTomotive Open System ARchitecture (AUTOSAR) and Portable Operating System Interface (POSIX). The following bullet points describe briefly how Automotive Ethernet affects the two approaches in the design process. For more information on AUTOSAR see Sections 3.5.3 and 5.4 or [5].

- **AUTOSAR:** Like for all operating systems, one of the core functions of AUTOSAR is to decouple the software from the hardware it is used on. For this, AUTOSAR provides a set of Application Programming Interfaces (APIs) that very specifically fit the automotive requirements and are very scalable. If desired, AUTOSAR can be used with very small 8-bit processors; although today in automotive the use of 32-bit processors is standard. Furthermore, AUTOSAR divides the software into modules which then, at least theoretically, can be developed independently by various companies. A mostly automated, powerful configuration tool chain combines the function modules. The result is a specific software project, which is partially postbuilt configurable. AUTOSAR is the dominant architecture in the body, chassis, and power train domains within a car.

  AUTOSAR supports Ethernet-based communication – first for DoIP – since 2009. The combination of AUTOSAR with SOME/IP (see Section 5.4) requires the extension of the configuration tool chain, in order for it to cover SOME/IP plus the respective "Socket Adaptors" and Service Discovery (SD) configuration. The first

AUTOSAR standard to describe these requirements is version 4.0.3. Further revisions of AUTOSAR incorporate more requirements coming from Automotive Ethernet (see Section 3.5.3 for more details). The Japan Automotive Software Platform and ARchitecture (JASPAR) also adopted the AUTOSAR standard.

- **POSIX:** POSIX compatible Operating Systems (OS) like QNX (see Section 3.2) or GENIVI (see Section 3.5.3) are mainly used in the infotainment domain. However, with the growing complexity in the driver assist domain, interest in deploying POSIX systems is increasing there, too. In contrast to traditional in-vehicle networking systems, Automotive Ethernet is the ideal complement to the POSIX approach; the socket-based communication of TCP/IP fits perfectly to the InterProcess Communication (IPC) solutions of the POSIX compatible operating systems. The Ethernet stacks are generally an integral part of these solutions. For the support of SOME/IP, extensions for code generators exist. Generally, an independent tool chain feeds these generators. An example of such a tool chain is based on an Interface Definition Language (IDL) named Franca, which is also used for development tools like Eclipse.

6.3 The Networking Architecture

6.3.1 EE Architecture in Perspective

From a consumer or telecommunications industry perspective it looks very complex to introduce a new in-vehicle networking technology. Why is it not possible to simply take an existing CE technology and (extensively) reuse it in cars? The simple answer is that CE technologies generally do not fulfill essential automotive requirements. One of the key differences between the IT/consumer and automotive industries emphasized in this book so far has been the physical environment the technology has to cope with.

As a minimum requirement, ambient temperatures of –40°C to +105°C have to be supported. For example, the inner side of the wing mirror of a car parked in the sun quite quickly heats up to this upper bound. In the engine compartment or the gearbox control, the temperature that needs to be supported increases even up to 125°C. This not only requires that the respective qualification programs are passed for the semiconductors, but also that the right type of housing is selected and that concepts for heat dissipation are included in the design. Next to the temperature range, mechanical strain, EMC, ESD, etc. also significantly differ in a car from an office environment. In consequence, Automotive Ethernet not only requires special semiconductor qualification but also optimized PHY technologies.

There are more differences than just temperature requirements, largely due to the different use case. Just to give an idea: A typical car, and thus also its communication network and power system, drives 300 000 km in 15 years, facing 10 500 temperature changes [6]. Additionally, a user might start and park a car several times a day. The design of the communication network and power supply system thus has to consider the following effects:

- **Long lifetime:** Cars are used for many years, and around the world are even getting older [7]. This touches an important user requirement: The customer wants a reliable and robust car with minimal maintenance. In consequence, a car manufacturer has to **consider the aging effects** of all components inside the car, i.e., of active as well as passive parts like, e.g., capacitors. The temperature changes and long activity cycles are particularly challenging. The PCB design, the selection of electronic parts, and the design of the communication system have to take this into account. Furthermore, the long lifetime affects the supply chain. Suppliers have to guarantee the **availability of (replacement) parts** for a significantly longer time than is usual in the consumer or IT industries. A minimum of 15 years is standard. This is not different for Automotive Ethernet than for any other in-vehicle networking technology. However, it might be new to traditional Ethernet suppliers.
- **Upgradeability:** The long lifetime of cars directly implies another aspect: Car owners generally keep their cars for several years (see, e.g., [8]). It is therefore desirable to be able to update the car, especially in the infotainment and consumer interface domain. This might be done by exchanging ECUs to newer versions or by upgrading the software of a system. Furthermore, a buyer of a second-hand car might want to add features the previous owner did not care for. The possibility of adding functions later therefore increases the resale value of a car. In some cases, updates can be realized via software update or ECU exchange. In other cases, the desired function can only be added with an additional ECU. In order for this to be possible, the originally installed in-vehicle network would need to have been designed to allow for these later extensions. The implications are quite different for the different in-vehicle networking technologies. In case of Automotive Ethernet one existing ECU in the network would need to have a switch and a currently not used PHY in order to allow a new ECU to be integrated into the network. To provide such an unused PHY in a network, however, adds costs (see also Section 6.3.2.2). Smart new concepts for ensuring the upgradeability thus have to be developed for Automotive Ethernet.
- **Long pauses between use:** Even if a car has been parked, i.e., was "off," for a long time, it needs to be able to start when the customer wants to use it again. With the increasing amount of electronics inside the car, this is a challenge as electronics consume power and thus drain the car's battery, even when the car is not in use. Some units need power because they always have to stay alive, like the ECU that enables keyless entry. Others simply consume quiescent current. One measure is to disconnect the battery from the ECUs when the car is not used (see also Section 6.3.3). Another is to **limit the quiescent current** of all electronic hardware (directly connected to the battery), i.e., to have (almost) **no power consumption when the car is not used**. The allowed quiescent current for a complete ECU that always stays on the power supply is in the area of a few 100 μA, which means the transceivers can be in the range of 10–20 μA quiescent current. In the consumer and IT industries, Ethernet semiconductors do not have to provide such low quiescent current values. Even if a device has not been used for a long time and the batteries are discharged, the user simply charges the batteries or connects the units to the mains. The consequence for automotive use of Ethernet is that ECUs with standard Ethernet components will consume too much

quiescent current. Respective ECUs thus cannot stay connected to the battery when the car is not in use or other intelligent means for just keeping the parts connected to the power supply that are needed in the off status have to be provided. This is one of the many criteria that need to be considered in the EE architecture.

- **Low power (petrol) consumption when the car is in use:** This requirement as such is not so different from the IT or consumer industries. All users would like small electricity bills, long battery life, or stand-by times. To ensure low energy use in all devices is therefore a common concern in all electronics. The difference inside a car is that a car combines a large number of functions from completely different domains. Not all functions provided inside a car are needed all the time, and so for the power budget it is advantageous if a function that is currently not needed does not use any energy. From the user's perspective, this will become more urgent with the proliferation of electric cars, in which drivers might actively switch off the entertainment or even air-conditioning system, in order to be able to drive some additional kilometers. From a car manufacturer's perspective, this is also urgent in order to meet government requirements in terms of CO_2 reductions for cars with combustion engines.

  The principle that describes controlling the on/off of some of the units via the in-vehicle network is referred to as "**partial networking**" [9]. Section 6.3.3 covers this in more detail; here are just some basic comments. There are different ways to deactivate an ECU not needed. The most energy-efficient one is to take a unit off the power supply (as is the case when the car is parked, see bullet point above). Sometimes, this might not be completely possible though and a unit needs to stay in wake-up mode, with the wake-up circuit/transceiver powered. This circuitry has the same low quiescent current requirements as above: 10–20 μA. If this cannot be reached, which is likely to be the case for Automotive Ethernet transceivers, "wake-on LAN," i.e., waking a unit up by sending a "magic" wake-up packet, is not ideal in an Automotive Ethernet network. Also, an ECU might, for a while, only be needed for forwarding data through its switch. Thus, only the switch semiconductor inside the ECU needs to stay alive. This is also not ideal because of the power budget of Ethernet parts. So, deciding where to place switches in respect to partial networking, is yet another criterion that needs to be taken into consideration when deciding on the EE architecture including the partial networking layout.

  It is by no means evident to decide which ECU can be deactivated in a partial networking situation. The acoustic park distance control is a good example to illustrate this. The acoustic signal is generally generated in the multimedia system, i.e., the HU. If this was completely taken off the power supply, because the user preferred not to have any entertainment, the park distance control would no longer work. For this reason, it makes sense to install coordination that controls the activation and deactivation of certain functions.
- **Fast availability:** In the consumer electronics domain, it is normal that consumers wait, e.g., for the availability of a mobile phone after complete switch off. Even to switch on a TV takes today almost as long as it did during the time of tubes. However, inside a car the user expects all functions to be available immediately. This is the case when the car is being started, and it is also the case when ECUs have been put

to sleep during the runtime of the car for the power saving reasons discussed above. Customers not be aware that this is even happening. Once the car has been started, it needs to be able to go. But even to start the engine, the in-vehicle network needs to be available, as, e.g., the electronic immobilizer system requires communication with various ECUs to exchange and calculate the respective certificates before the engine can start. Because of the previously discussed requirement for low quiescent current it is not possible to simply leave all needed ECUs connected to the power supply. This leads to the requirement of a **short start-up time**. After 100–200 ms the in-vehicle network needs to be ready to communicate, in order not to cause any noticeable delay in the wake-up of the systems behind. However, for the complete multimedia or navigation system these short start-up times can often not be reached, and a user might have to wait a few seconds (see Table 5.2 in Section 5.1.2.4).

- **On–off changes:** An Ethernet-based IT network expects not to face extreme temperatures or even a lot of temperature variations. Additionally the whole network will be shut down only occasionally for maintenance reasons and equipment upgrades. It contrast, an in-vehicle network is started and shut down frequently, often several times during one day. Short start-up times are thus crucial (see above and also Section 5.1.3). This requirement is the same for Automotive Ethernet as for all other in-vehicle networking technologies. Additionally, the concepts behind the start-up are important. For Automotive Ethernet they vary in comparison to the traditional bus in-vehicle networking technologies (see also Section 6.3.3).
- **Quality and availability of power supply:** An Ethernet-based IT network is expected to be constantly and evenly powered. Even though a low electricity bill will also be of some concern in an IT environment, the risk of a power supply with significantly varying voltage or the risk of running out of power altogether is small and if it happens, it is generally part of a more serious problem out of the network provider's control. In cars, this is different. The power supply is always limited. When the car is parked for a long time, the limit is the battery capacity and charge status. When the car is running, the limit is the amount of petrol in the tank (in combination with the battery charging capability). Additionally, low-voltage impulses might occur in the case of old batteries, low temperatures, and engine starts. This can cause resets, which in return results in the necessity of fast recovery. Furthermore, the power needed will vary. Depending on the actual use of the car, more or fewer functions are connected to the network and the power supply. In the service case, even only one ECU might be powered. Without giving any further details on how these issues are addressed in the EE architecture – all car manufacturers have different requirements and capabilities in this respect – they have to be taken into account when designing in-vehicle networking and using Automotive Ethernet.

6.3.2 The In-Vehicle Communication Network

6.3.2.1 Integrated versus Distributed EE Architecture

During the system design, there are two principal approaches on how to partition function blocks: an integrated and a distributed one. In the integrated approach, many

Table 6.1 Integrated versus distributed EE architecture

Integrated EE architecture	Distributed EE architecture
Fewer, more complex, and bigger ECUs	More, less complex, and smaller ECUs
Fewer connections in the in-vehicle network, which potentially makes it less complex	More connections in the in-vehicle network, which potentially makes it more complex
More or less the same functions for all	Better scalability of overall car functionality (potentially over different models)
Fewer suppliers, which might not be the best for all integrated functions	For each ECU a different, an optimum supplier can be chosen
Likely lower costs if all functions are always paid for	Likely higher costs if all functions are always paid for

function blocks are designed into one ECU, while in the distributed approach only a few function blocks are designed into one ECU.[3] This means that to achieve the same functionality, there are fewer ECUs and a smaller communication network in the integrated than in the distributed approach. Provided the customer is willing to pay for the functionality, this generally makes the integrated approach less costly. At the same time, the integrated approach is less flexible and less scalable. Everyone is offered the same functions. Even if some of the functions are activated by software, so that it is still possible to sell functions as options, every customer will have the same hardware provisions in the car. Table 6.1 provides an overview of the differences.

Overall, one approach is not necessarily better than the other. The preference also depends on the market segment a car manufacturer wants to address, either with a specific car model or in general. As a rule of thumb, integration is more common for basic functions, for functions that have been on the market for a long time, which have higher take rates, and/or for cars and functions in the lower price segment. Distribution is more likely if the function is an option, is new, and/or addresses the top end of the market. Naturally, there are other criteria for partitioning functions on ECUs than just the general preference on an integrated or distributed architecture, e.g., the physical or logical closeness of functionalities [1].

In respect to understanding Automotive Ethernet the difference between integrated and distributed architectures is important. First, the choice impacts the need for Automotive Ethernet, i.e., the more distributed a network is, the more units need to communicate and be connected via an IVN technology. Thus the more likely that Ethernet is a choice. In some cases the availability of Ethernet might even become a prerequisite to separate functions, when this would result in communication with the higher data rates that Ethernet supports. This leads to the second aspect. Automotive Ethernet simplifies handling a distributed architecture. It allows for scalability in the data rate by exchanging the PHY technology (only) and supports various different option configurations over time and different car models in a cost optimized way (see Section 6.3.2.5).

6.3.2.2 From a Bus to a Switched Network

The traditional in-vehicle network technologies (CAN, LIN, FlexRay, MOST; see Section 2.2) are all bus systems, meaning that the available bandwidth is shared between

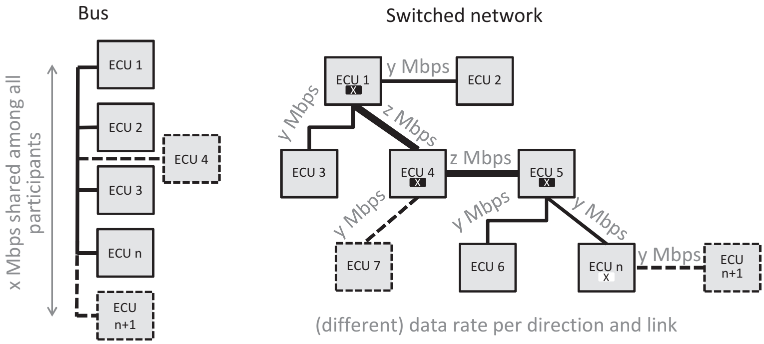

Figure 6.2 Important topology and extendibility differences between a bus and a switched network. Added units are marked by dotted lines. The boxed "x" marks units with switches, whereas the white box indicates that here a switch would need to be added.

all units connected. Most often, a bus system has a line topology; with MOST being an exception, using a ring topology. Owing to the simplex nature of the optical transmission MOST uses, closing the line to a ring saves having to use two POFs between neighboring units. Instead, it allows all units to communicate with each other, even though communication is in one ring direction only.

A fundamental property of a bus is that, in principle, all units can listen to and thus receive all data that is available on the channel. There are different ways to decide whether a receiver actually does process the data and the traditional in-vehicle networking technologies do deploy a variety. There can be a message identifier, from which a receiver decides whether a data is of interest (CAN). They can follow a predefined schedule (LIN, FlexRay). There can be an address unambiguously identifying the receiving unit (like in MOST).

If more units need to be included in the communication on the bus, it seems straight forward to simply attach a new unit to the bus (see also Figure 6.2). Such a layer 1 connection requires that there is enough data rate available to support the traffic for and from new unit. Repeaters work on layer 1 and might be used to extend the range of a bus. Hubs might be used to connect various users at one point. In the case of FlexRay a star coupler is needed when the network increases such that the propagation delays of the signals become too large. Additionally, on layer 2 it needs to be ensured that the new unit is included in the channel access scheme, which might require reprogramming of all units that share the same bus (e.g., in case of FlexRay). In case of MOST, the order of units needs to be observed.

Should there be reasons why a new unit cannot be connected to the same bus, e.g., because the data rate of the bus is not sufficient, a proxy/gateway is needed to transfer data from one bus to the other. Traditionally, this gateway function between two busses of the same technology is performed on the Network or Transport Layers. However, because of the timing requirements and the availability of the star couplers, gatewaying between two FlexRay busses is not so common. For MOST, it is also unlikely. Not only because of the complexity (see also Section 3.1.2.1) but also because the use case is simply not so probable. MOST focuses on the infotainment domain, which is a typical example of an application area in which many new technologies evolve (causing distribution), but also in which they become legacy quickly (and are integrated). For example, to begin with, digital radio or digital TV functions were provided with separate ECUs. With the advent of software defined radio, the digital radio and TV functionalities are now offered integrated onto the standard radio chip. This in return can make it more cost efficient to provide the respective hardware in all cars and to enable the functions by software if the customer selected them. Consequently, for car manufacturers that follow this approach, there are two ECUs less to consider in the infotainment domain.

Communication from one CAN bus to another is also not that obvious, as this would require sharing message IDs. Extending an existing CAN bus would be easier. If this is not possible because of bandwidth limitations, the easiest way for a gateway to handle this is to simply pass a packet "as is" from one bus to the other. A more generic approach is to unpack the content and depending on the message, repackage it onto a new bus, which then might be FlexRay, LIN, MOST, or even Ethernet as well. Naturally, this second approach is more resource demanding, causing more latency, too. In general, gateways require effort, which increases with the time criticality of the applications. Communication within one networking technology is thus preferable.

This often leads to a so-called domain architecture; with the idea that most communication happens within one domain. However, the data rate provided by a certain bus might not be sufficient. Additionally, it is not always possible to avoid cross-domain traffic. When thinking of new concepts like automated driving – which require a significant amount of redundancy – this is obvious. But there are many examples a lot simpler. Displaying the picture of a rear view camera on the head unit requires cross-domain communication between the driver assist domain and the infotainment domain.

With Automotive Ethernet the basic concepts of the architecture are completely different. First, each connection is Point-to-Point (P2P) meaning that always ever only two units are attached to the same link segment. The networking happens on layer 2, on which switches pass data on, depending on addresses (in Figure 6.2 the switches are marked with an "X"). This is key for networking but completely new in automotive. It supports all kinds of topologies and no gateways are needed from Ethernet to Ethernet, even if various different PHY technologies and speed grades are used (for examples see Section 4.3). Proxies/gateways are only needed between Ethernet and other in-vehicle networking technologies.

Extending the network requires adding another port to a switch or exchanging a PHY with a two port switch (see also Figure 6.2). Because the links are always P2P this actually increases instead of decreases the overall capacity inside the network. Of course, the limits of the available data rate also need to be observed when extending an Ethernet-based network. It is better for the overall data rate to attach a unit directly or close to the main communication partner and not at the other end of the network. However, with the data rate so much higher to start with – per link and direction 100 Mbps – and technologies for even higher data rates being developed (see Section 4.1.3), the right architecture should always be able to provide enough bandwidth. This in return gives more flexibility for network optimization. This potentially allows prioritizing network optimization criteria differently.

6.3.2.3 Unicast, Multicast, and Broadcast in a Switched Network

As was explained above, every unit attached to a bus will, in principle, see all packets transmitted on that bus and the bus therefore has natural broadcast. It is up to the reception concept whether the channel is used broadcast-like, i.e., everybody reads all packets, or not. In a switched Ethernet network, this is different. Using P2P with only ever two units directly connected, the core transmission mode of a switched Ethernet network is unicast. However, also an Ethernet system supports multicast and broadcast for specific use cases and/or with the use of the specifically reserved MAC addresses.[4] Also, a switch performs broadcast every time it receives a message for which it does not know to which port to forward it to. This is naturally frequently the case during start-up, when the network is used for the first time. Another example for multicast are 1722 audio video packets, which are multicasted within the AVB cloud (see Section 5.1).

The obvious risk of broadcast and multicast messages is the overload of the network, especially if the network has redundant paths. In theory, the redundancy should be taken care of with the Spanning Tree Protocol (STP), but there is a remaining risk, not only of a high data load but also of malfunctions. Even though there is a theoretical difference between broadcast and multicast, switches generally just flood both types of packets to all outgoing ports except the one the packet came from [10]. An exception are AVB capable switches, which make a hardware supported distinction between ports connecting to other units that are part of the AVB cloud and ports connecting to units which are not.

In Automotive Ethernet another effect is important. A PHY in an Ethernet-based network receives every packet it detects on the link attached to it and passes it to whatever processing capability containing the MAC it finds attached to its MII interface. In an end node this is likely to be a μC; in a middle node this is a switch. An end node μC will continue processing every packet that contains its own address, every broadcast message, and probably, depending on the implementation, most of the multicast messages. Because there is very limited filtering at the MAC level, it is up to the application to decide what to do with the content. If all packets in an Ethernet network were broadcast or multicast packets, e.g., because a CAN bus was emulated over the Ethernet network, the application on the μC would look at the message content of every packet before it could be dismissed. The μC of a small ECU, designed to process, e.g., 10 Mbps of data, would be congested just by the network functionality.

In an Automotive Ethernet network, it is thus better to avoid the use of broadcast and multicast. Under the assumption that an Automotive Ethernet network will always be small (in comparison with an IT network), fanouts, i.e., the use of multiple unicast transmissions instead of one multicast transmission, are preferable. If done with care, this is not only less strenuous for the network but also scales well, in case the system grows.

6.3.2.4 Throughput Optimization with Automotive Ethernet

Crucial for designing the communication network is that all data arrive at the receiver within the required timeline. In order to ensure this, it is first necessary to know all communication paths between units and the amount of data to be transmitted. Normally, this is one of the outputs from the EE architecture and the communication architecture

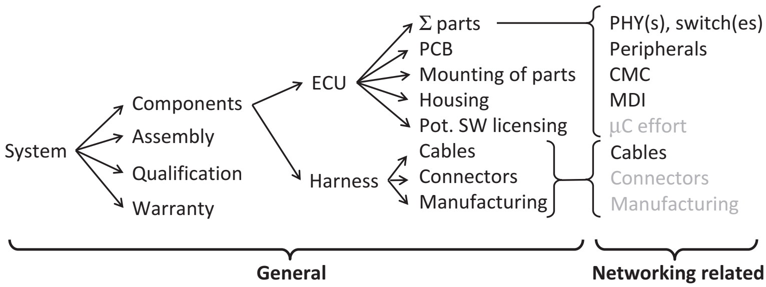

Figure 6.3 Overview of cost elements. Light gray indicates the elements needed for the competitive comparison between technologies but not needed for the topology optimization [1].

then decides on the in-vehicle networking technology to use, how many gateways to include, and where/how to connect the units to the network in a way that it scales over option selection, car models, and functional updates. Functional updates are generally offered to customers once a year, but of course their requirements are not known when the car is being designed to start with. The in-vehicle network thus has to allow for future growth in data rate requirements.

For designing the communication network, tools and simulations are used, especially to determine how much of the available data rate is used up. This is not obvious to do, as the communication rarely follows deterministic and/or cyclic patterns. Deterministic approaches do exist, but often result in the establishment of maximum possible traffic demand or occurrence of bursts. Deterministic approaches thus can be deployed to determine the required buffer space in an ECU.

The rules behind the communication network design that allow the determination for maximum loads, prioritization, and maximum delays, are not specific to Automotive Ethernet. Traditional in-vehicle networking technologies need them as well and they are thus well established in automotive. Examples of existing tools evaluating in-vehicle network load performance include PREEvision (Vector), SystemDesk (dSpace), SymTA/S, and TraceAnalyzer (Symtavision) (see also [11]).

Naturally, the tools need(ed) to be extended with the communication paradigms of Ethernet. To compute reserves for the traditional technologies is very complex in comparison and "higher" data rates even require the inclusion of the wave theory. The simulation of the bus physics of an Ethernet network has to take the switched architecture into account. Between switches there are P2P links only and the computation of the traffic load on those links is comparably straightforward. The critical design element is the load inside the switches. The in-vehicle network design has to ensure that a certain switch in the network does not become the bottleneck because of too small buffers. Depending on the tooling preferences within the car manufacturers, the respective tools have been/need to be extended accordingly, including AVB.

6.3.2.5 Cost Optimization in a Switched Network

Costs inside a car comprise of various elements (see Figure 6.3, or, e.g., [1]). The obvious cost elements are the costs for ECUs and harness as well as the costs associated with

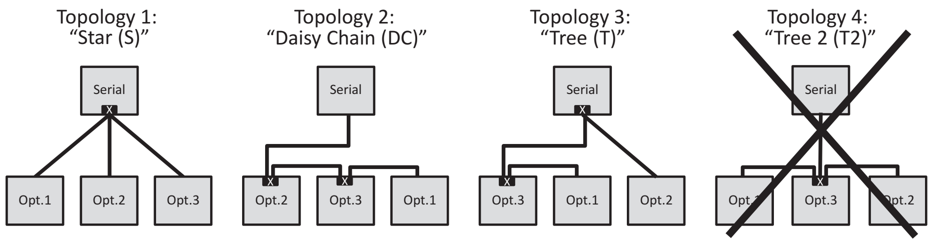

Figure 6.4 Example Ethernet topologies with four ECUs.

assembling the hardware into a car. To derive sound values for these elements is daily business within every car manufacturer and an integral part when deciding on ECUs or the in-vehicle networking system to use. Additionally, every ECU or in-vehicle networking technology also causes costs in qualification and later warranty. However, those are hard to tag at the time of decision and are thus generally not considered this early in the process. Even more difficult are those aspects that have monetary impacts but are difficult to measure like the money saved by being more flexible or future-proof with a certain architecture or the marketing effect achieved with offering a certain function. These aspects, important as they are, are thus not (yet) included as hard values in the cost assessment.

In respect to Automotive Ethernet, two types of cost related evaluations can be expected. The first is a direct comparison between Automotive Ethernet and another networking technology. Most likely Automotive Ethernet will be compared with technologies with similar use cases like MOST150 or pixel links. In this case, all costs directly related to the choice of the networking technology are added up (see also Figure 6.3): cabling, connectors, harness manufacturing, transceiver chip costs, switches, peripherals, filters, CMC, SoftWare (SW) licensing fees,[5] and potentially additional effort in the μC, e.g., for compression or providing an MII interface (or for providing the MLB interface in the case of MOST, see Section 2.2.4.2).

The second evaluation is related to the actual Ethernet topology. Because Ethernet is not a bus but a switched network with P2P links, criteria new to the industry need to be included in the optimization. This section will focus on the explanation of those, also, because a correctly optimized Ethernet network can make a big difference in the cost comparison between two competing technology solutions.

The switch is the main element that is new in the optimization. The switch allows for flexibility when selecting the topology, but also adds costs, and the location of switches thus needs to be considered carefully. The following will describe the effects with a simple example of four "Ethernet ECUs," of which one is available in all cars and three are offered as options. For this example it is assumed that the customers can select each option individually, i.e., there are no complement selections required or mutual exclusions of options. Figure 6.4 shows the three topologies that are in principle possible under this assumption. Key is that all topologies have enough PHYs and switch ports available to support all possible subsets of options selected. The "Tree 2" (T2) topology would not allow for this.[6] If the customer chose only the outside two options and not the

Table 6.2 Basic comparison values for the different topologies (left) using the example link distances on the right

part	S	DC	T	[m]	Serial	Opt. 1	Opt. 2	Opt. 3
# PHYs	3	2	2	Serial	0	4	2	1
# 2port[a] switch	0	2	2	Opt. 1		0	6	3
# 3port[a] switch	1	0	0	Opt. 2			0	3
# segments	3	6	4	Opt. 3				0
Max. Σ lengths	7	8	6					

[a] Two/three integrated PHYs. The switch will have (at least) one internal port. This is why a "2port switch" is sometimes called a "2+1port switch" or even a "3port switch."

one with the switch, the serial ECU would not have enough ports available to connect them. For the Daisy Chain (DC) and the Tree (T) topologies there are various variations of (optional) ECU order possible. Figure 6.4 shows one example of each topology. When comparing the different types of topologies it makes thus sense to first select the optimal ECU order for each. In the example calculated in the following, the topology internal optimization was completed upfront and is not explained in the text. It is a fundamental requirement for all topologies that the required data rate is supported by all of them, which also needs to be confirmed before making the cost calculation. The numbering of the optional ECUs chosen in Figure 6.4 represents the optimum selection for the example discussed below.

There are some principal differences between the three topologies independent of the specifics of a certain car model: the number of PHYs and switches each solution needs, and the number of harness segments that need to be defined per topology (see Table 6.2). Each harness segment describes one possible element of a harness a car manufacturer has to define for the harness manufacturing, depending on the options selected. If in the star topology Options 2 and 3 are not selected, nothing changes for the connection to Option 1. The overall number of link segments is three. If in the daisy chain Options 2 and 3 or only Option 3 are not selected, Option 1 requires a new link segment in order to be connected to the network. The overall number of segments is thus six. Table 6.2 compares the different topologies. The maximum length of cabling in the topologies requires the data from a specific car model. For the values shown in the left-hand part of Table 6.2 the distances between the units as defined in the right-hand part were assumed.

At first glance, the DC topology looks less favorable than the T topology, as both the summed up cable lengths as well as the number of link segments that need to be defined are larger for DC than for T. However, in T one of the switches is in the serial unit, i.e., in every car, independent from how many options the customer buys. Without knowing more details about the take rates of the options and their take rate combinations, it is not possible to say which topology is the most suitable. For the star scenario it is important to know, for example, how often the customers choose how many of the options, i.e., how often only one, two, or three ports are needed in the switch. This determines how often one, two, or all three ports of the switch are not needed and thus represent an unnecessary hardware provision. Note that it is not an option for the car manufacturers

Table 6.3 Example of average networking-induced costs per vehicle, depending on topology and car model take rates

	Star	Daisy Chain	Tree
Model A (high end)	9.27	9.75	9.37
Model B (upper class)	7.70	7.34	7.30
Model C (middle class)	4.91	2.78	4.04
Volume-dependent average	5.44	3.64	4.66

Note: The input data for this table is given at the end of the chapter.

to simply make three variants of an ECU, each with a network interface of a different size (switch with three ports, two ports, or PHY only).

A car manufacturer can only handle a certain number of variants for each ECU especially in logistics and assembly, but also in development, qualification, and purchasing as too many variants are problematic. Often, there are important functional reasons to have them. For example, a HU needs to have different variants for different countries/regions. On top, like other ECUs, a car manufacturer might want to offer a "light," "mid," and "full" version. If each of those versions then needed to be provided in versions with different Ethernet interfaces, this would quickly become unsustainable.

Table 6.3 shows the result of cost calculations for three different cars. Model A is a high-end car, Model B an upper class car, and Model C a middle class car.[7] The table shows that with a simple cost estimation, for each of them a different topology would be optimum. If the goal is to support only one topology in all car models, in this example the DC topology is the most cost-efficient one, simply because the switch, which is a costly hardware element, is needed only when the respective option is selected. Note that a topology optimization might include other criteria, like extendibility, discovery, or power consumption (see Section 6.3.3). In a daisy chain, it is more effort for the serial ECU to discover which options have actually been selected than in a star topology. Also, the costs for the additional harness segments have not been included. If too unfavorable from a cost perspective, it might be better to have a solution with as few link segments as possible.

Naturally, the example was very simplified and a real life example is likely significantly more complex. Its purpose was to explain the additional elements needed to optimize an Automotive Ethernet Architecture, in comparison with traditional in-vehicle networking technologies.

6.3.3 The Supply Network

Most car designs have the following three different types of power circuits, which are distinguished by the "clamp" the circuits and units are connected to. For example, units on "Clamp 15" receive power only when the engine has been switched on. Units on "Clamp 30" are directly connected to the battery and always have power. "Clamp 30g" is also connected to the battery but switched. Additionally, there might be function specific solutions like controlling the power supply of the cameras attached to a Surround View System (SVS) directly by the SVS, as this ensures that the cameras are only active when needed. The SVS in return is a good example of a special situation, as it needs to be powered only when the reverse gear is used.

The power circuits represent the different status a car can be in. The car can be parked, with only very few systems needing to be powered like the alarm or door lock systems (Clamp 30). When the engine is on and the customer is driving, most systems are on (Clamp 15). Then, there are the special situations like the above mentioned reverse driving, which triggers the power of the SVS. Another scenario is the situation when the car stopped with the engine switched off, but with passengers still using some of the functions. A driver might want to finish a telephone call he or she is having via the in-built hands-free system or the kids want to continue watching the movie in the Rear Seat Entertainment (RSE) while the parent buys petrol, etc. All these functions might be switched separately on the power supply (Clamp 30g). Which function is on which power supply can vary from car manufacturer to car manufacturer, especially in respect to those functions that are provided when the engine is off. Electric window opening, mirror adjustments, and GPS use are good examples for functions that are handled differently by different OEMs and which are sometimes available when the car has been unlocked, sometimes when the key has woken up the car, and sometimes when the engine has been started.

The key purpose of the power supply network is to reliably provide power whenever needed. Next to the energy efficiency that will be discussed in more detail in this section, "reliably" includes that the power is supplied at a constant level without voltage drops, which might make ECUs go into reset. One of the most critical times for voltage drops is the engine start, but other situations are possible.

The less power is required the easier it is to provide sufficient power. This is especially critical for the quiescent current of those units that are on Clamp 30, i.e., that always receive power, even if the car is parked (see also Section 6.3.1). Customers expect their car to start also after a long time of not using it. However, how much quiescent current is needed depends mainly on the ECU design and available semiconductors. The use of Automotive Ethernet has some impact on this but not much a designer can change. In contrast, having units use as little power as possible during the runtime of a car is a consequence of choices made during the development of a car.

Having ECUs use less power during the runtime of the car generally means putting them to sleep when they are not needed and waking them up when they are needed, potentially in a certain order. "Sleep" in this context can mean various things: sleep of the complete ECU, sleep of parts of the ECU, or cutting off the power completely. Normally, it is decided at the application level that an ECU is not needed for a while and should go into sleep. When using Automotive Ethernet, this does not change. However, the designers might have to consider that an ECU, while not needed for its core function, might be needed for its Ethernet switch. So, either a unit that is not continuously needed during the runtime of a car should not be designed to contain a switch, which is continuously needed to forward network traffic (see also Section 6.3.2.5) or the unit needs to be designed such that the switch is excluded from the ECU's sleep mode. Even with Ethernet transceivers not being particularly energy efficient, there might be other architectural reasons for this being a better choice than routing the traffic via another ECU.

Wake-up allows for three design choices: Wake-up can be performed via (a certain) activity on the communication link; via a separate wake-up line and pin or circuitry (continuous signal or pulsed); or via the power supply, which would require intelligent power supply with individual fuses and switches. For CAN a methodology has been developed to wake up nodes via a certain wake-up signal on the data line. Note that even though all systems in which only parts of the network units are active perform "partial networking," often only the data line activity detection method as used by CAN is referred to as partial networking [9]. However, decisive for this concept is that CAN represents a bus system with a shared media. The method has to support that only some units on a bus are asleep, while others on the same bus remain active. In a switched Automotive Ethernet network the challenge is different. As has been discussed in Section 4.4.6, a wake-up notification needs to propagate through switches.

Note that Energy-Efficient Ethernet (EEE) does define a signal, which reactivates Ethernet links previously in energy-efficient mode (see Section 4.4.3). EEE aims at reducing the power consumption in the Ethernet transceiver. Partial networking as such aims at saving power of the complete ECU. This means that the existing EEE approach is a good start but not sufficient.

Last, but not least, sleep mode and wake-up method impact latency, QoS, the wake-up/start-up time and there are interrelations with the topic of Power over Data Line (PoDL) (see also Section 4.4.3), when the power supply is used for wake-up. Obviously, there are some choices for the designers and from a standardization perspective; it leaves room for work to be done in the future.

6.4 Test and Qualification

For any new in-vehicle networking technology, the availability of suitable tools is decisive and it is necessary to ensure the availability of tools very early in the process. After all, tools need to be tested too, and the pilot project discussed in Section 3.4.2 was used also for this purpose with some of the output reflected in [11]. Tools are needed for supporting the development (see the previous sections) as well as for test and qualification, which this section will focus on. This includes tools used during the long time cars need to be maintainable once they have left the production site.

Like for the development tools, not everything needed to be redone from scratch when testing and qualifying Automotive Ethernet. At the same time, some reuse is possible from the IT industry exploiting the ISO/OSI layering separation of Automotive Ethernet. An example is the tool Wireshark, which allows the universal tracing of IP-based communication. In the following, each test step shown in Figure 6.1 is discussed separately, highlighting the differences caused by Automotive Ethernet:

- **Component-level testing:** Tests at the component level evaluate the functionality of an ECU as a standalone component. However, many functions provided by ECUs require data from the in-vehicle communication network. In order to test the functionality of an ECU as a standalone component it is thus necessary that the communication data is available during testing without needing the other ECUs. With parts being supplied by various different Tier 1s this is essential for the development of a car. In automotive, tools for "rest/residual bus simulation" are used for this. These tools generate all data the ECU requires from the communication system. The introduction of Automotive Ethernet required substantial extensions to these tools. Not only are the communication patterns as such different, some of the communication paradigms change as well. It is no longer sufficient just to simulate the output data of other ECUs and provide it as input data to the ECU being tested. An Automotive Ethernet ECU needs to be registered as a new participant in the overall communication network. The ECU interacts with the partners in terms of service discovery or other method calls (see also Section 5.3). The rest bus simulation tools need to support all this.

  Additional to enabling the communication with ECUs not physically available, the output of the ECU under test onto the communication network needs to be tracked and evaluated for correctness and completeness. Naturally, the respective tools need to support Automotive Ethernet communication. To aid testing on ECU level, an Ethernet ECU test specification was made available by the OPEN Alliance [12].
- **Network-level testing:** These tests evaluate all functions that are directly related to the communication network. In the first step, the communication behavior of each unit is tested stand alone for these functions with help of the rest bus simulation described in the previous bullet point. In the second step several ECUs are connected. The component test of the previous step and the integration tests of the consecutive test both test the (customer) functions of the ECU. These are excluded in the network-level testing, unless they are necessary for locating erroneous behavior of an ECU in the network functions. This distinction is important, because without knowing that the communication network as such functions correctly, it makes no sense to test the communal functionality of the ECUs in the integration tests. Tests performed at the network level comprise, e.g., start-up and restart, shutdown and sleep, load tests under expected circumstances, load test in the case of bursts, and behavior in the case of network overload.

  On the tool side this requires tools that generate the respective network traffic and loads and naturally, these have to support Automotive Ethernet in order to be useful with an Automotive Ethernet network. However, as Ethernet PHYs and switches need to be tested outside of the automotive world as well, the change and challenge are small with tools and experience available. Furthermore, tools for data logging and evaluation are needed, which can be the same as for the component-level tests. The change in the case of network tests for Automotive Ethernet is not the tools as such, but the way of adding the tools to the network (see Figure 6.5). When logging the data on a bus, the logger can be just another participant of that bus (left-hand diagram in

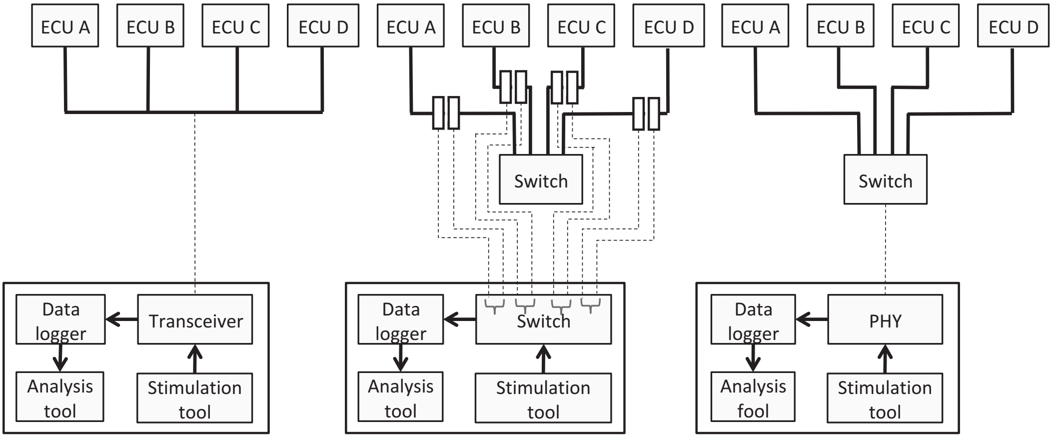

Figure 6.5 Data logging on a bus (left side) and in a switched network (middle and right side).

Figure 6.5). In a switched Ethernet network, however, the traffic flow on every link can be different. To track all communication in a complex Ethernet network either each link needs to contain a transparent logger (middle diagram in Figure 6.5) or each switch has to provide an extra port that allows mirroring the communication to the data logger attached (right-hand diagram in Figure 6.5). With the costs of Ethernet ports, such a "mirroring port" adds costs to the ECUs. Breaking each link, requires more complex test setups and impacts the timing behavior of the Ethernet communication. Whether the timing impact matters or not depends on the exact use case, and is up to the tester to decide.

Tools for logging an Automotive Ethernet network have been made available. One other aspect that needs consideration in this context is the amount of data that is being logged. This is twofold. First, with video being analog or on distinct LVDS links, there was no temptation to log all video data in a network. With Ethernet, there suddenly is. Within the shortest amount of time this can lead to tremendous amounts of tracking data. It is important to have the possibility to distinguish in the data logging between control traffic, which is likely the focus of the tests, and potentially dismiss the application data. Second, if the middle setup shown in Figure 6.5 is used and, e.g., ECU A communicates with ECU D, the same content will be recorded twice. Once it is clear that no packets are dropped and that switching errors are out of discussion, it might be desirable to avoid the redundancy.

- **System-level testing:** These tests investigate the functionality of the integrated system, either of the complete system or of subsystems, i.e., functional domains. The focus is on testing the customer functions against the requirement specifications, with all (respective) ECUs available. To handle the complexity, test automation is important, independent from any in-vehicle networking technology used. Additionally, the above discussed tooling can be used to log the communication data in case erroneous behavior needs to be evaluated.

- **Final in-vehicle acceptance test and maintenance:** The system tests are an important step toward the overall functionality, but, of course, all required functions have to pass the final in-vehicle acceptance test with all units integrated into the vehicle. This is again independent from the in-vehicle networking technology chosen. However, once a car has been qualified for Start of Production (SOP), the need for tests and tools has not ended. The difference is that after SOP tests are performed only in the event of malfunctions in the field. Key for solving the customer issues in the field, are the diagnostic capabilities designed into the ECUs to start with.

  These diagnostic functions follow two different approaches. First, if the component registers an anomaly considered during the development, the component will record the respective error code in its error memory. With reading the error code from this memory the malfunction is known to the repair personnel. Additionally, there are various test functions available to locate the source of a malfunction better. As explained in Section 2.1 diagnostics was one of the very early use cases for in-vehicle networking.

  A good example is the electric window opener. If a customer complains of a defective window opener, the repair personnel will first read the error memory of the electronics inside the window opener. The error code might say that the motor of the window opener ceased functioning because the motor temperature was too high. The error code does not say whether the cause for it was a defect in the electric motor, or whether the mechanical run of the window was hampered. To find out, a diagnostic function of the external tester could stimulate the opening and closing of the window, while receiving data on the forces on the window. If the forces are in a defined value range, it will be necessary to change the electric motor. If the forces are in a different range, it will be necessary to repair the mechanics and gaskets inside the door. This example describes a classic repair situation.

  Nevertheless, sometimes a problem arises that cannot be solved by the above methods. The specialists needing to investigate the issue have to have access to all information sources available, including the in-vehicle network. So, even if this is a rare situation, it is essential that the in-vehicle network traffic is also accessible in customers' cars and that the respective tools are capable of deciphering Ethernet traffic.

Notes

1 An example of a bottom-up test strategy in a narrower context was presented in Section 4.1.3, where the EMC was first tested on semiconductor, then on ECU, and then at car level.

2 Note that this is a very ECU/hardware centric approach that describes the prevailing car development process at the time of writing. With the increase of software and software enabled functions in the car, it is unlikely that this process will sustain unchanged. The interrelations and -dependencies will be just too complex (see also Chapter 7).

3 This applies not only to the EE architecture but actually also to semiconductor design. On top both the partitioning onto semiconductors and the partitioning onto ECUs are mutually dependent. If a semiconductor is available that integrates two function blocks it is more likely

that an ECU will too. The integrated chip might make it financially attractive. If there is no obvious market for integrating those two blocks in one ECU, e.g., because the take rate for one of the function blocks is low, such a chip is not necessarily offered. This makes it an interesting chicken and egg problem, as the integrated chip might reduce the price of the function block such that the take rate would increase. However, even though semiconductor availability is very important for automotive innovations, it is not the focus of this chapter.

4 The Ethernet broadcast address is FF:FF:FF:FF:FF:FF. Ethernet MAC addresses for multicast are identified by a “1” in the least significant bit of the first octet. A list of Ethernet multicast addresses can be found in [13].

5 SoftWare (SW) as a cost factor is comparatively new in the automotive industry as such. Generally, the costs for the SW development are seen as a one-time effort the Tier 1 vendor has to deal with during the development of the ECU. Implementing a different business model based on SW licensing is rare. However, it can be expected that the automotive industry will see more of it in the future. Today, explicitly disclosed SW licensing costs are generally caused by a standard from the consumer industry being reused inside the vehicle (e.g., specific media compression formats). Note that car manufacturers expect hardware related licensing fees to be part of the hardware offer and to be dealt with by the Tier 2 (and thus Tier 1) supplier and to be transparent to the car manufacturer.

6 For the four-unit scenario shown, the T2 topology is actually a star topology with the center shifted to a different unit. However, the center is shifted to an option. To emphasize the distinction to the S scenario, and to emphasize the possible addition of more optional units, this is called “Tree 2.”

7 The following shows the input data used to derive the values of Table 6.3. While the take rates would represent high-end (Model A), upper class (Model B), and middle class (Model C) cars, neither they, nor the cost values have been taken from a real life example, but were selected to emphasize the differences. There is no point in including any low-end car in such a comparison as the number of options chosen for those is generally very small. For the cost values, the following, arbitrarily picked example values have been used:

- Standalone PHY: 1.00
- Switch logic: 0.90
- Integrated PHYs: 0.70
- CMC and Media-Dependent Interface (MDI): 0.25
- Cable per [m]: 0.35

One-time costs like the number of segments have not been included, as this would require absolute cost and volume values to do so. The table below shows the combination take rates used for the example.

Opt. 3	Opt. 2	Opt. 1	Serial	Model A	Model B	Model C
no	no	no	yes	0%	15%	60%
no	no	yes	yes	3%	10%	20%
no	yes	no	yes	1%	4%	7%
no	yes	yes	yes	10%	28%	7%
yes	no	no	yes	1%	1%	3%
yes	no	yes	yes	10%	2%	2%
yes	yes	no	yes	5%	10%	1%
yes	yes	yes	yes	70%	30%	0%
				0.1 Mio	1.0 Mio	5.0 Mio

References

[1] T. Streichert and M. Traub, *Elektrik/Elektronik-Architekturen im Kraftfahrzeug*, Berlin: Springer Vieweg, 2012.

[2] Euro NCAP, "Euro NCAP," continuously updated. [Online]. Available: www.euroncap.com/en. [Accessed 9 September 2016].

[3] Wikipedia, "V-Model (software development)," 27 July 2016. [Online]. Available: http://en.wikipedia.org/wiki/V-Model_(software_development). [Accessed 9 September 2016].

[4] ISO, "ISO 26262–1–10:2011 Functional Safety," ISO, Geneva, 2011.

[5] O. Kindel and M. Friedrich, *Softwareentwicklung mit AUTOSAR: Grundlagen, Engineering, Management in der Praxis*, Heidelberg: dpunkt, 2009.

[6] VDE, "ITG-Positionspapier: Kfz-Anforderungen an Elektronik-Bauteile," ca. 2005. [Online]. Available: www.vde.com/de/InfoCenter/Seiten/Details.aspx?eslShopItemID=ffc61ce5–392e-41fa-aa98-e885c3ea68f4. [Accessed 25 January 2014].

[7] R. Charlton, "American Drivers Keeping Cars on the Road for Longer: Average Age Now 11.4 Years," 9 August 2013. [Online]. Available: www.huffingtonpost.com/reno-charlton/american-drivers-keeping-_b_3718301.html. [Accessed 7 February 2014].

[8] kbb.com, "Average length of U.S. vehicle ownership hit an all-time high," 23 February 2012. [Online]. Available: www.kbb.com/car-news/all-the-latest/average-length-of-us-vehicle-ownership-hit-an-all_time-high/2000007854/. [Accessed 7 February 2014].

[9] all-electronics.de, "Deutsche OEMs setzen Standards," 14 June 2011. [Online]. Available: www.all-electronics.de/deutsche-oems-setzen-standards/. [Accessed 17 December 2013].

[10] Wikipedia, "Multicast address," 7 September 2016. [Online]. Available: http://en.wikipedia.org/wiki/Multicast_address. [Accessed 9 September 2016].

[11] M. Schaffert and D. Kim, "Automotive Ethernet Tooling Revision 13," 12 August 2016. [Online]. Available: http://opensig.org/tech-committees/tc4/. [Accessed 9 September 2016].

[12] OPEN Alliance TC8, "OPEN Alliance Automotive Ethernet ECU Test Specification," OPEN Alliance, Irvine, 2016.

[13] cavebear.com, "Multicast (including Broadcast) Addresses," 3 September 1999. [Online]. Available: www.cavebear.com/archive/cavebear/Ethernet/multicast.html. [Accessed 19 January 2014].

7 Outlook

There are in principle two types of innovations: incremental and revolutionary/radical.[1] While incremental innovations improve key market features on the basis of existing methods and technologies, radical innovations are based on new technological developments that either significantly change an existing market and/or even create a new one. The uncertainty of the consequences of an innovation increases from incremental to radical. For example, the costs and improvements for an incremental innovation can generally be assessed more easily than for a radical innovation. A radical innovation often requires more investment and holds a greater risk of technological failure. The important aspect, however, is that a radical innovation has the chance to change a market so profoundly that – among other consequences – it will leave behind those who did not perform the change (see, e.g., [1]).

At the time of writing, there were two automotive innovation fields with potentially far-reaching consequences: alternative drives/electric vehicles and automated driving. For both innovation fields, networking is extremely important, and both require the vehicle to be connected to the worldwide network. Furthermore, both have the potential to change the car related user behavior significantly more than, e.g., the introduction of navigation systems did. For example, with the limitations of current battery technologies, the reach of an electric car is shorter than that of a car with a combustion engine. Because of the shorter reach, it is extremely important to be able to correctly predict the remaining reach at any time. Users will become aware of the difference it makes if they switch off the air conditioning or infotainment and they might develop new behavior patterns. This includes wearing gloves, opening the windows, or accepting a detour if this avoids very hilly terrain.

The same applies to automated driving. The general car occupancy rate in the developed countries is well under 1.5 [2]. This means that during a significant amount of drive time only one occupant, the driver, is inside the car [2]. So even if there are cars that have been designed as family cars, the design focus for the vast majority of functions is still on driving and the driver. Today, drivers need to pay attention to the road, not only with their eyes, but they also need to keep their hands on the wheel and keep their minds focused. There are good reasons why mobile phone use is seen as so critical inside cars, even with hands-free provisions (see, e.g., [3]).

Automated driving will not be introduced in one go, neither in terms of the functions that are automated nor in terms of the roads on/velocities at which these functions can be used. Cars will need to be designed for drivers for a long time to come. However, with every extra minute the driver does not have to pay attention to the driving, the use and function of the car will change. Obviously, TV and video entertainment will experience a boost, but who knows what else will change? Potentially, the complete system of traffic management, vehicle buying behavior or health care will require rethinking (less stress). It is part of an innovation being radical that not all consequences are known upfront.

The introduction of Ethernet in automotive is also a radical change. At the time of starting this effort, only few people realized the dimension of the change. Even today, the introduction of Ethernet remains for some nothing more than the introduction of a new in-vehicle communication technology. But, Automotive Ethernet is similar to automated driving in that it is at the same time an innovation in itself and an enabler for innovations of significant proportion. For the reader of this book it should be obvious that with Automotive Ethernet a bigger conversion in the automotive market is taking place, where in the end the Ethernet solution sets the grounds for a modern communication network. The following describes in more detail the important changes Automotive Ethernet brings to in-vehicle networking and that it empowers not only the in-vehicle networking but the EE architecture as such.

Traditional in-vehicle networking operates with many restrictions, in terms of bandwidth, packet size, costs, weight, and higher layer protocols. The more experienced the designer, the more he or she will design along those restrictions. This automatically limits the creativity as the designers immediately discard ideas that would only be possible if it were not for the restrictions. Ethernet-based communication lifts some of these restrictions; in terms of bandwidth, in terms of networking as such, in terms of networked functions and higher layer communication. The networking aspect is very important as within the switched network it provides flexibility while eliminating the need to take proxies and gateways into consideration.

One of the fundamental changes of Automotive Ethernet is in its use as a switched Point-to-Point (P2P) network instead of as a shared bus (see, e.g., Chapter 6). As has been explained, this opens new possibilities for the in-vehicle communication architecture that have so far been impractical from a functional perspective. For example, a bus generally requires a line topology within one domain complying with the data rate constraints of the used technology. A switched network, in contrast, allows any type of topology with the maximum data rate available on every link. Optimization can then be done on the basis of criteria like weight, location, or space, if there is desire to do so. This in return might relax the design and restrictions in harness manufacturing.

Another fundamental change with Ethernet-based in-vehicle networking that has been addressed extensively in this book is the reuse and flexibility along the ISO/OSI layering. An obvious example is the PHY layer. Section 4.3 describes various PHY technologies, which include different speed grades, pixel links, and different media like copper or optical transmission or even wireless.[2] All can be used via switches in the same network that are transparent for the application. Being able to choose from different PHY technologies is noteworthy not only because of the variety but also because of the technical capability. 15 years ago, electrical in-vehicle networking systems were being replaced by optical systems. At that time it was thought that electrical communication systems could not support data rates in the multi-Mbps range in the automotive environment (see Section 2.2.4). In 2016, electrical transmission of multi-Gbps Ethernet was being discussed for the same use (see Section 4.3.3.1). This catch up of "knowledge" sets the basics for a cost optimized, switched network with full-duplex communication layer instead of simplex ring structures at fixed bandwidth.

Ethernet-based communication induces significant changes also on layer 2 beyond switching. For example, for automated driving, safety is of utmost importance. In order to provide safety, it is crucial to have various forms of redundancy available, both in the network (in case a link is disrupted) as well as in the information sources (in case one of the sources fails). More than one unit should combine and use the data. It needs a powerful networking technology to be able to provide enough full-duplex data rate as well as redundancy concepts. Because of being switch based an Ethernet-based IVN network can grow in bandwidth. The standard developments around Time-Sensitive Networking (TSN) allow for different redundancy concepts (see Section 5.1.4). TSN is a good example of how Ethernet grows and changes with requirements. Automated driving is a good example of a use case for which the massive scalability and underlying modularity of Automotive Ethernet are advantageous.[3]

Furthermore, Ethernet-based communication suggests the use and development of open standards on all layers of the ISO/OSI layering model. In 2016, when Automotive Ethernet was still in the introduction phase, such developments required initialization. But at some point in the not too distant future, the number of Automotive Ethernet ports produced per year is likely to catch up with the number of ports produced for the IT industry. Taking 70 million cars produced annually and making the conservative assumption that every car will have on average only three Ethernet links, i.e., uses six Ethernet ports, it is expected that this will soon be achieved (see Table 1.3 in Section 1.3). Such numbers will encourage the development of protocols that support all kinds of automotive applications, especially if open platforms are used. This is one example on how the automotive world can reuse modern, sophisticated and contemporary solution from the "modular" Ethernet ecosystem.

The ecosystem Ethernet-based communication provides thus goes beyond layer 1 and 2. Future vehicle architectures, as, e.g., needed for automated driving, have to provide scalable platforms that do not only allow to design complex and distributed functions, but that set up a stable framework that can be filled with constantly more functions. It is important to be able to reduce a vehicle system to its basic functions, which can then be put together as a modular Lego concept. The possible scaling is very critical to car manufacturers, as is the possibility to reuse the same function in very different vehicle platforms and generations. Timeliness of vehicle platform software updates – such as the customer is used to in other industries – is a common goal for all car manufacturers. At the same time, there will be some fixed elements, e.g., suspension, engine, and sensors, in the vehicle architecture that are very difficult to update during the vehicle life time. For this reason stable interfaces to these fixed elements are extremely important [4].

The foundations of respective future EE architectures will be based on three essential building blocks [5]:

- **Service-Oriented Architecture (SOA):** It changes the need for the today dominant thinking in ECUs, i.e., in their functions and partitioning, as it establishes a new paradigm of services available (anywhere) in the system. Ethernet-based communication directly supports SOA with SOME/IP (see Section 5.4).
- A unit that provides a database, fiduciary services and **message broker concepts**: In traditional communication systems, essential information on, e.g., a CAN bus is directly modeled onto the respective CAN bus. In future designs, this essential information might need to be available in other domains or even outside the car, too, where the receiver would have a hard time handling a CAN message. The content of the CAN message could, for example, be needed in the backend system of the car manufacturer. Ethernet communication and IP addressing, in contrast, are well established also outside the cars.
- **Powerful computing platforms:** These will be the home for highly complex algorithms. Artificial intelligence approaches will have an enormously important role in case of automatic driving applications. These powerful platforms have to interact with the rest of the car and the outside infrastructure. IP-based communication systems will be state of the art for such platforms and high-bandwidth, full-duplex Ethernet communication will be needed to exchange data in between.

Ethernet is an indispensable pillar of the future IVN system, since all the necessary communication concepts for the blocks described are perfectly embedded in Ethernet, just as the systems have been interlocked in the IT industry. It is quite realistic that with the introduction of Automotive Ethernet the in-vehicle network will no longer be a limiting factor for automotive functions, but an enabler. This transition, however, will be slow. As the introduction of Automotive Ethernet takes time, its proliferation takes time, too, while it can be expected that parts of the in-vehicle network will not transition to Ethernet. In the end a slow approach is good and necessary. There are many engineers participating in the whole chain, who need to learn and adapt, without challenging the customers' safety on the way.

What has been described in this book is just the start [6]. What the final result will be, once the full properties of the technology have been exploited, cannot be fully assessed now. Only time will tell. The biggest challenge will be to strictly stick to the separation between the layers, whatever the changes and innovations are. The biggest opportunity is to be able to think "outside the box" of in-vehicle networking restrictions in order to design the automotive future without them. The biggest luck is to be part of it.

Notes

1 A third categorization often used (and often confused) for innovations is "disruptive" [7]. Disruptive innovations are often similar in their outcome to that of revolutionary/radical innovations: They can significantly change the market. However, the reason and the way this happens

is different for radical innovations than for disruptive innovations. A radical innovation overtakes the market from the left (which is the expected way for overtaking when you drive on the right side of the road like in continental Europe), while a disruptive innovation overtakes the market from the right (i.e., in an unexpected way in continental Europe). Radical innovations overtake a market by significantly improving key features (e.g., faster processing, more throughput, more load, longer battery life) and serving key customers. This is what disruptive innovations do NOT do. Instead they exploit other properties of an innovative new technology that, at least to begin with, do nothing to improve the key features and might even be inferior (e.g., less processing, less throughput, less load, shorter battery life). Instead, other properties are emphasized (e.g., size, ease of use, being usable in additional use cases, . . .). Over time, consumer behavior and expectations might actually change and they value those other properties more, with the result that companies having successfully followed the traditional path suddenly loose out. Disruptive innovations are thus difficult to forecast. While disruptive innovations are an important concept managers should keep in mind, the concept and distinction between radical and disruptive innovations is of less concern for this book. This book does therefore not explored the disruptive or nondisruptive elements of Automotive Ethernet. At first glance, Automotive Ethernet first of all provides more of something in-vehicle networking always needed more of: data rate. It thus seems an unsuitable candidate for a disruptive technology.

2 One possibility that has not been discussed so far is the inclusion of wireless links into the network. In automotive up to now, WLAN– IEEE 802.11 technologies to be precise, which are sometimes called "Wireless Ethernet" (see, e.g., [8]) – are mainly being discussed in the context of consumer device integration and Car-to-Car (C2C)/Car-to-anything (C2X) communication. The main application areas for the latter are traffic flow and traffic safety improvements (e.g., [9] [10]). Consumer device integration is a necessity in terms of customer comfort (see also Section 2.2.7). However, all IEEE 802.11 technologies integrate seamlessly into an Ethernet-based network, so why not use them for the car internal communication, too? After all, wireless communication promises a maximum reduction of cable weight. Good reasons for now have been EMC, costs, and the need for power supply. However, with the establishment of Ethernet-based communication in automotive, WLAN offers itself as one more PHY technology in the Ethernet-based network, also for in-vehicle communication.

3 The authors have no particular opinion on whether the introduction of Automotive Ethernet will actually help to increase the pace of innovation in the automotive industry, nor whether this is actually productive. The pace of innovation, which apparently some customers require [11], has other boundary conditions than just the in-vehicle networking technology, like the design cycles of the complete car. What can be expected to change with Automotive Ethernet is simply the range of innovations that are less limited by networking boundary conditions.

References

[1] V. Kotelnikov, "Radical Innovation versus Incremental Innovation," not available. [Online]. Available: www.1000ventures.com/business_guide/innovation_radical_vs_incr .html. [Accessed 3 January 2014].

[2] International Energy Agency and OECD, "Saving Oil in a Hurry," IEA Publications, Paris, 2005.

[3] D. Harper, "WATCHDOG REPORT: Statistics Show Hands-Free Cell Phone Use While Driving Is Just as Dangerous," 23 May 2010. [Online]. Available: www.pressofatlanticcity .com/news/top_three/article_dcab06b6–6614–11df-9e65–001cc4c03286.html. [Accessed 5 January 2014].

[4] T. De Doete, "Welcome Speech: Exploiting automotive Ethernet to accelerate product innovation," in *IEEE-SA (6th) Ethernet&IP@Automotive Technology Day*, Paris, 2016.

[5] K. Barbehön, "Mitten im Umbruch zur Digitalisierung: Wie die Zukunft der E7E Architektur von IT-Standards beeinflusst wird," in *21st International Congress on Advances in Automotive Electronics*, Ludwigsburg, 2016.

[6] T. Königseder, "Automotive Ethernet 12 years old: Time for feedback and lessons learned," in *IEEE-SA (6th) Ethernet&IP@Automotive Technology Day*, Paris, 2016.

[7] C. M. Christensen, *The Innovator's Dilemma*, New York: HarperBusiness, 1997.

[8] E. Geelen, "Insights from Metro Ethernet Forum's Americas Summit in San Diego," 30 January 2013. [Online]. Available: www.tellabs.com/blog/index.cfm/2013/1/30/Insights-from-Metro-Ethernet-Forums-Americas-Summit-in-San-Diego. [Accessed 5 January 2014, no longer available].

[9] K. Matheus, R. Morich, I. Paulus, C. Menig, A. Lübke, B. Rech, and W. Specks, "Car-to-Car Communication – Market Introduction and Success Factors," in *5th European Congress on Intelligent Transportation Systems*, Hannover, 2005.

[10] K. Matheus, R. Morich, and A. Lübke, "Economic Background of Car-to-Car Communication," in *Symposium Informationssysteme für mobile Anwendungen*, Brunswick, 2004.

[11] T. Grünweg, "Eine Industrie kommt auf Speed," 10 February 2013. [Online]. Available: www.spiegel.de/auto/aktuell/warum-lange-entwicklungszyklen-fuer-autohersteller-zum-problem-werden-a-881990.html. [Accessed 15 February 2013].

www.cargeek.ir

www.cargeek.ir

Index