

Document Title	Requirements on Ethernet Support in AUTOSAR
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	419

Document Status	published
Part of AUTOSAR Standard	Classic Platform
Part of Standard Release	R21-11

Document Change History			
Date	Release	Changed by	Description
2021-11-25	R21-11	AUTOSAR Release Management	<ul style="list-style-type: none"> • Validation of 10BASE-T1S • Validation of Ethernet Wake on data line
2020-11-30	R20-11	AUTOSAR Release Management	<ul style="list-style-type: none"> • Introduction of 10BASE-T1S (DRAFT) • Introduction of Ethernet Wake on data line (DRAFT)
2019-11-28	R19-11	AUTOSAR Release Management	<ul style="list-style-type: none"> • Introduction of Ipsec • Changed Document Status from Final to published
2018-10-31	4.4.0	AUTOSAR Release Management	<ul style="list-style-type: none"> • Introduction of Transport Layer Security - TLS (DRAFT)
2017-12-08	4.3.1	AUTOSAR Release Management	<ul style="list-style-type: none"> • Editorial changes
2016-11-30	4.3.0	AUTOSAR Release Management	<ul style="list-style-type: none"> • Improvements of switch related requirements • Introduction of testing and diagnostics features • Editorial changes
2015-07-31	4.2.2	AUTOSAR Release Management	<ul style="list-style-type: none"> • Editorial changes

2014-10-31	4.2.1	AUTOSAR Release Management	<ul style="list-style-type: none"> • Introduction of IPv6 for in-vehicle communication • Support for Global Time Synchronization over Ethernet • Support for Switch Control/Configuration, Semi-Static Auto-Configuration • TcpIp generic upper layer support (CDD)
2013-10-31	4.1.2	AUTOSAR Release Management	<ul style="list-style-type: none"> • Added support for ISO 13400-2 • Updated Document Traceability • Editorial changes
2013-03-15	4.1.1	AUTOSAR Administration	<ul style="list-style-type: none"> • Introduced IPv6 Protocol, VLAN • Added support for SWS_Service_Discovery • Added support for SWS_TCPIP • Added support for SWS_DoIP • Editorial changes
2013-03-15	4.1.1	AUTOSAR Administration	<ul style="list-style-type: none"> • New traceability mechanism
2010-09-30	3.1.5	AUTOSAR Administration	<ul style="list-style-type: none"> • Initial Release

Disclaimer

This work (specification and/or software implementation) and the material contained in it, as released by AUTOSAR, is for the purpose of information only. AUTOSAR and the companies that have contributed to it shall not be liable for any use of the work.

The material contained in this work is protected by copyright and other types of intellectual property rights. The commercial exploitation of the material contained in this work requires a license to such intellectual property rights.

This work may be utilized or reproduced without any modification, in any form or by any means, for informational purposes only. For any other purpose, no part of the work may be utilized or reproduced, in any form or by any means, without permission in writing from the publisher.

The work has been developed for automotive applications only. It has neither been developed, nor tested for non-automotive applications.

The word AUTOSAR and the AUTOSAR logo are registered trademarks.

Table of Contents

1	Scope of Document	6
2	Conventions to be used	6
2.1	Document Conventions	6
3	Acronyms and abbreviations	7
4	Requirements Specification	8
4.1	Functional Overview	8
4.1.1	TCP/IP Protocol Stack (TCPIP)	8
4.1.2	SWS Socket Adaptor (SoAd)	8
4.1.3	SWS Diagnostics over IP (DoIP)	9
4.1.4	SWS Ethernet Interface (EthIf)	9
4.1.5	SWS Ethernet Driver (Eth)	9
4.1.6	SWS Ethernet Transceiver Driver (EthTrcv)	9
4.1.7	SWS Ethernet Switch Driver (EthSwT)	9
4.1.8	SWS Ethernet State Manager (EthSM)	9
4.1.9	SWS UDP Network Management (UdpNm)	9
4.1.10	SWS Service Discovery (Sd)	10
4.2	General Requirements	10
4.2.1	Configuration	10
4.2.2	Operation	10
4.2.3	AUTOSAR Interfaces	11
4.3	Functional Requirements	12
4.3.1	SWS TCP/IP Protocol Stack	12
4.3.1.1	TCP/IP General Requirements	12
4.3.1.2	TCP/IP Internet Protocol	13
4.3.1.3	TCP/IP Sub Module: IPv4	14
4.3.1.4	TCP/IP Sub Module: IPv6	15
4.3.1.5	TCP/IP ICMP	19
4.3.1.6	TCP/IP Sub Module: ICMPv4	20
4.3.1.7	TCP/IP Sub Module: ICMPv6	20
4.3.1.8	TCP/IP TCP	20
4.3.1.9	TCP/IP TLS	22
4.3.1.10	TCP/IP UDP	25
4.3.1.11	TCP/IP DHCP	25
4.3.1.12	TCP/IP Sub-Module: DHCPv4	26
4.3.1.13	TCP/IP Sub-Module: DHCPv6	27
4.3.1.14	TCP/IP IPsec	28
4.4	SWS Socket Adaptor (SoAd)	30
4.5	SWS Diagnostics over IP (DoIP)	36
4.6	SWS Ethernet Interface (EthIf)	39
4.7	SWS Ethernet Driver (Eth)	41
4.8	SWS Ethernet Transceiver Driver (EthTrcv)	44

4.9	SWS Ethernet Switch Driver (EthSwt)	49
4.10	SWS Ethernet State Manager (EthSM)	52
4.11	SWS UDP Network Management (UdpNm)	54
4.12	SWS Service Discovery (Sd)	54
4.13	Non-Functional Requirements (Qualities)	55
4.13.1	SWS TCP/IP Protocol Stack	55
4.13.2	SWS Socket Adaptor (SoAd)	55
4.13.3	SWS Ethernet Interface (EthIf)	55
4.13.4	SWS Ethernet Driver (Eth)	56
4.13.5	SWS Ethernet Transceiver Driver (EthTrcv)	56
4.13.6	SWS Ethernet State Manager (EthSM)	57
4.13.7	SWS UDP Network Management (UdpNm)	57
5	Requirements Tracing	58
6	References	62

1 Scope of Document

This document is intended to list the general requirements resulting from concept UID 387 TCP/IP CommStack Extensions (TCP/IP).

These will need to be implemented by the following software specifications:

- 414 UDP Network Management (UdpNm)
- 415 Ethernet State Manager (EthSM)
- 416 Socket Adaptor (SoAd)
- 417 Ethernet Interface (EthIf)
- 418 Diagnostics over IP (DoIP)
- 430 Ethernet Driver (Eth)
- 431 Ethernet Transceiver Driver (EthTrcv)
- 616 Service Discovery (Sd)
- 617 Tcp/Ip Stack (TcpIp)
- 656 Ethernet Switch Driver (EthSwt)

2 Conventions to be used

2.1 Document Conventions

The representation of requirements in AUTOSAR documents follows the table specified in [TPS_STDT_00078], see Standardization Template, chapter Support for Traceability ([1]).

The verbal forms for the expression of obligation specified in [TPS_STDT_00053] shall be used to indicate requirements, see Standardization Template, chapter Support for Traceability ([1]).

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as follows.

Note that the requirement level of the document in which they are used modifies the force of these words.

- **MUST**: This word, or the adjective "LEGALLY REQUIRED", means that the definition is an absolute requirement of the specification due to legal issues.
- **MUST NOT**: This phrase, or the phrase "MUST NOT", means that the definition is an absolute prohibition of the specification due to legal issues.

- **SHALL:** This phrase, or the adjective "REQUIRED", means that the definition is an absolute requirement of the specification.
- **SHALL NOT:** This phrase means that the definition is an absolute prohibition of the specification.
- **SHOULD:** This word, or the adjective "RECOMMENDED", means that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
- **SHOULD NOT:** This phrase, or the phrase "NOT RECOMMENDED", means that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.
- **MAY:** This word, or the adjective "OPTIONAL", means that an item is truly optional. One vendor may choose to include the item because a particular market-place requires it or because the vendor feels that it enhances the product while another vendor may omit the same item.

An implementation, which does not include a particular option, SHALL be prepared to interoperate with another implementation, which does include the option, though perhaps with reduced functionality. In the same vein an implementation, which does include a particular option, SHALL be prepared to interoperate with another implementation, which does not include the option (except, of course, for the feature the option provides.)

3 Acronyms and abbreviations

The glossary below includes acronyms and abbreviations relevant to SRS Ethernet that are not included in the AUTOSAR Glossary [2].

Abbreviation / Acronym:	Description:
ARP	Address Resolution Protocol
COTS	Commercial Of The Shelf
DAD	Duplicate Address Detection
DEM	Diagnostic Event Manager
DET	Default Error Tracer
DHCP	Dynamic Host Configuration Protocol
DHCPv4	Dynamic Host Configuration Protocol for Internet Protocol Version 4
DHCPv6	Dynamic Host Configuration Protocol for Internet Protocol Version 6
DoIP	Diagnostics over IP





Abbreviation / Acronym:	Description:
HTTP	HyperText Transfer Protocol
IANA	Internet Assigned Numbers Authority
ICMP	Internet Control Message Protocol
ICMPv4	Internet Control Message Protocol for Internet Protocol Version 4
ICMPv6	Internet Control Message Protocol for Internet Protocol Version 6
IETF	Internet Engineering Task Force
IP	Internet Protocol
IPv4	Internet Protocol for Version 4
IPv6	Internet Protocol for Version 6
MTU	Maximum Transmission Unit
NDP	Neighbor Discovery Protocol
SoAd	AUTOSAR Socket Adaptor Module
TCP	Transmission Control Protocol
TCP/IP	A family of communication protocols used in computer networks
TLS	Transport Layer Security
UDP	User Datagram Protocol
UdpNm	AUTOSAR UDP Network Management Module
XCP	eXtended Calibration Protocol

4 Requirements Specification

This chapter describes all requirements driving the work to define the Requirements on Ethernet Support in AUTOSAR.

4.1 Functional Overview

4.1.1 TCP/IP Protocol Stack (TCPIP)

The TCP/IP protocol stack (TCP/IP stack) is intended to handle layers 2 through 4 of the ISO/OSI layer model. This includes, but is not limited to protocols like IPv4, IPv6, DHCPv4, DHCPv6, ARP, NDP, TCP, UDP, ICMPv4, ICMPv6, TLS and others.

In Terms of this model the SoAd and therefore the whole AUTOSAR COM stack above represent the application at layer 7.

4.1.2 SWS Socket Adaptor (SoAd)

The SoAd is an adaptor layer, not only matching the AUTOSAR APIs to standard socket APIs [3], but also mapping PDU IDs to socket connections.

4.1.3 SWS Diagnostics over IP (DoIP)

The DoIP part of the SoAd implements the functionality required by [4].

4.1.4 SWS Ethernet Interface (EthIf)

The Ethernet Interface provides standardized interfaces to provide the communication with the Ethernet bus system of an ECU. The APIs are independent from the specific Ethernet Controllers and Transceivers and their access through the responsible Driver layer. The Ethernet Interface is conceptually able to access one or more Ethernet Drivers and Ethernet Transceiver Drivers via one uniform interface.

4.1.5 SWS Ethernet Driver (Eth)

The Ethernet Driver offers uniform interfaces for the Ethernet Interface. The Ethernet Driver hides hardware specific details of the used Ethernet controller.

4.1.6 SWS Ethernet Transceiver Driver (EthTrcv)

The Ethernet Transceiver Driver offers uniform interfaces for the Ethernet Interface. The Ethernet Transceiver Driver hides hardware specific details of the used Ethernet transceiver.

4.1.7 SWS Ethernet Switch Driver (EthSwT)

Ethernet is a switched network and switches need to be integrated into AUTOSAR-ECUs. These ECUs need to have the capability to configure Ethernet switches.

Ethernet Switch Driver provides an interface for managing and controlling a switch and its ports.

4.1.8 SWS Ethernet State Manager (EthSM)

The Ethernet State Manager offers uniform interfaces for the Communication Manager (ComM). The Ethernet State Manager hides network specific details.

4.1.9 SWS UDP Network Management (UdpNm)

The UDP Network Management offers uniform interfaces for the Network Management Interface (NmIf). The UDP Network Management hides network specific details.

4.1.10 SWS Service Discovery (Sd)

Service Discovery serves primarily to determine the operational status of a service. A service may be comprised of any combination of SW-Cs and/or BSWs in an ECU. The BswM is used to aggregate the SW-Cs and BSWs status and trigger Sd when the required combination is available. Secondly Sd can also be used as a registration protocol for publish/subscribe communication patterns. In both cases Sd can transport addressing information in addition to the registration and availability information.

4.2 General Requirements

4.2.1 Configuration

[SRS_Eth_00053] SWS shall specify configuration [

Description:	The SWS documents shall include a configuration section to allow adaption of the functionality.
Rationale:	The functionality needs to be adapted to different use-cases and environments.
Use Case:	–
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01056](#))

4.2.2 Operation

[SRS_Eth_00085] Robustness against the change of logical addresses [

Description:	The Ethernet-related BSW modules of the AUTOSAR communication stack shall be able to handle the change of logical addresses of remote nodes.
Rationale:	ECUs can be configured for dynamic address assignment (e.g. DHCP) hence it is possible that ECUs change their addresses
Use Case:	DoIP, V2G, SD
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01784](#))

[SRS_Eth_00111] Robustness against unexpected communication patterns [

Description:	The Ethernet-related BSW modules shall provide configuration options to restrict the processing of frames or PDUs that don't fulfill certain expected properties.
Rationale:	This allows to define data policies and to drop all unexpected traffic, thereby reducing the processing overhead. This also limits the impact of faulty communication and the vulnerability to malicious attacks.
Use Case:	Robustness
Dependencies:	–
Supporting Material:	–

](RS_BRF_01784)

[SRS_Eth_00112] Ethernet-related BSW modules shall report relevant runtime errors from the used protocols [

Description:	The Ethernet-related BSW modules shall collect relevant runtime error information from the used protocols and report them to the Default Error Tracer module.
Rationale:	This allows error handling and monitoring of erroneous behavior.
Use Case:	Error handling and monitoring
Dependencies:	–
Supporting Material:	[5] IETF RFC 792; [6] RFC 793; [7] RFC 1122; [8] RFC 4443

](RS_BRF_01784)

4.2.3 AUTOSAR Interfaces

[SRS_Eth_00055] SoAd shall support UDP NM [

Description:	The SoAd shall offer an interface API for UDP NM to send network management messages. This interface shall mimic the API of interfaces in Autosar.
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	–

](RS_BRF_01784)

[SRS_Eth_00056] SoAd shall support XCP [

Description:	The SoAd module shall offer a data path to the AUTOSAR XCP module.
Rationale:	Exchange XCP frames between master and client
Use Case:	Calibration and Measurement
Dependencies:	–
Supporting Material:	ASAM XCP

](RS_BRF_01656)

[SRS_Eth_00058] SoAd shall support generic upper layers [

Description:	The SoAd shall offer an interface API for a generic upper layer to send and receive PDUs via socket connections, to control socket connections and to provide notification of socket connection and IP address assignment state changes.
Rationale:	Simplifies adding of additional upper layers
Use Case:	DoIP, V2G, DNS-SD
Dependencies:	–
Supporting Material:	–

](RS_BRF_01056)

[SRS_Eth_00103] Tcplp shall support generic upper layers [

Description:	The Tcplp shall offer a socket-based API for a generic upper layer to send and receive data, to control sockets and to provide notification of sockets and IP address assignment state changes.
Rationale:	Simplifies adding of custom testing module for TCP/IP protocol testing allowing a more flexible testing than only using SoAd.
Use Case:	TCP/IP Protocol Testing during development and later phases
Dependencies:	–
Supporting Material:	–

](RS_BRF_01784, RS_BRF_01056)

4.3 Functional Requirements

4.3.1 SWS TCP/IP Protocol Stack

4.3.1.1 TCP/IP General Requirements

[SRS_Eth_00054] TCPIP minimum functionality [

Description:	If DoIP is not implemented, the minimum requirements on the functionality of the TCP/IP stack shall be determined by the configuration of the Socket Adaptor.
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01784](#))

[SRS_Eth_00019] TCP and UDP related requirement specified in IETF RFC 1122 shall be implemented [

Description:	TCP and UDP related requirements specified in IETF RFC 1122 shall be implemented in the TCP/IP stack
Rationale:	–
Use Case:	IETF RFC 768, IETF RFC 793
Dependencies:	–
Supporting Material:	[7] IETF RFC 1122

]([RS_BRF_01784](#))

[SRS_Eth_00129] The TCPIP shall support access to measurement counter values [

Description:	The TCPIP shall provide an optional API to access measurement counter values, like for example the number of dropped datagrams
Rationale:	–
Use Case:	Interface for diagnostics
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

4.3.1.2 TCP/IP Internet Protocol

[SRS_Eth_00045] TCPIP automatic IP address assignment [

Description:	The TCP/IP stack shall implement a mechanism to automatically configure an IP addresses.
Rationale:	This is necessary when no static IP-addresses are assigned to still allow for plug and play configuration of the TCP/IP stack.
Use Case:	–
Dependencies:	[SRS_Eth_00002]
Supporting Material:	[9] IEEE 802.1AS

](RS_BRF_01784)

[SRS_Eth_00115] TCPIP shall allow to categorize outbound IP traffic [

Description:	The TCP/IP stack shall implement a mechanism to assign labels and/or service classes to the outbound traffic.
Rationale:	This allows network components like routers to differentiate the traffic based on this information and treat it appropriately.
Use Case:	Improved traffic forwarding algorithms, monitoring
Dependencies:	–
Supporting Material:	[10] IETF RFC 2474; [11] IETF RFC 6437

](RS_BRF_01784)

4.3.1.3 TCP/IP Sub Module: IPv4

[SRS_Eth_00014] IPv4 shall be implemented according to IETF RFC 791 [

Description:	The Internet Protocol (IPv4) shall at least be implemented in the TCP/IP stack as stated in IETF RFC 791
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	[12] IETF RFC 791

](RS_BRF_01784)

[SRS_Eth_00015] ARP shall be implemented according to IETF RFC 826 [

Description:	For DoIP the Address Resolution Protocol (ARP) shall at least be implemented in the TCP/IP stack as stated in IETF RFC 826
Rationale:	–



△

Use Case:	–
Dependencies:	[SRS_Eth_00014]
Supporting Material:	[13] IETF RFC 826

](RS_BRF_01784)

[SRS_Eth_00022] The dynamic configuration of IPv4 link-local addresses as specified in IETF RFC 3927 shall be implemented [

Description:	For DoIP the dynamic configuration of IPv4 link-local addresses as specified in IETF RFC 3927 shall be implemented in the TCP/IP stack
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	[14] IETF RFC 3927

](RS_BRF_01784)

4.3.1.4 TCP/IP Sub Module: IPv6

[SRS_Eth_00059] IPv6 shall be implemented according to IETF RFC 2460 [

Description:	<p>Internet Protocol Version 6 (IPv6) shall at least be implemented in the TCP/IP stack as stated in IETF RFC 2460.</p> <p>Limitation: the Authentication and Encapsulating Security Payload extension headers and other IPsec functionalities may not be supported. The handling of IPsec headers shall be tolerated</p> <p>Limitation: The TcpIp shall limit the IETF RFC 2460 to support only the reception of IPv6 fragment header and forbid the transmission. IETF RFC 2460 section 5. Packet Size Issue discourage the use of IP Fragmentation and therefore it can be covered by IETF RFC 1981 Path MTU Discovery for IP version 6.</p> <p>Extension: The Deprecation of Type 0 Routing Headers in IPv6 shall be implemented as stated in IETF RFC 5095</p>
Rationale:	–
Use Case:	In-vehicle communication and diagnostics
Dependencies:	–
Supporting Material:	[15] IETF RFC 2460; [16] IETF RFC 5095; [17] IETF RFC 1981

](RS_BRF_01784)

[SRS_Eth_00089] The Deprecation of Type 0 Routing Headers shall be implemented according to IETF RFC 5095 [

Description:	The Deprecation of Type 0 Routing Headers in IPv6 shall be implemented as stated in IETF RFC 5095
Rationale:	–
Use Case:	In-vehicle communication and diagnostics
Dependencies:	[15] IETF RFC 2460
Supporting Material:	[16] IETF RFC 5095

]([RS_BRF_01784](#))

[SRS_Eth_00090] The Neighbor Discovery Protocol shall be implemented according to IETF RFC 4861 [

Description:	The Neighbour Discovery Protocol shall be implemented in the TCP/IP stack according to the host specification of IETF RFC 4861 unless stated otherwise.
Rationale:	IETF RFC 4861 replaces the IETF RFC 2461
Use Case:	In-vehicle communication and diagnostics
Dependencies:	–
Supporting Material:	[18] IETF RFC 4861

]([RS_BRF_01784](#))

[SRS_Eth_00110] The Relationship between Links and Subnet Prefixes shall be considered according to IETF RFC 5942 [

Description:	The Neighbour Discovery Protocol implemented in the TCP/IP stack shall use the definition of the relationship between links and subnet prefixes as stated in IETF RFC 5942.
Rationale:	IETF RFC 5942 updates IETF RFC 4861
Use Case:	In-vehicle communication and diagnostics
Dependencies:	–
Supporting Material:	[19] IETF RFC 5942;

]([RS_BRF_01784](#))

[SRS_Eth_00113] The Neighbour Discovery Protocol shall provide a configuration possibility for Prefix List entries [

Description:	The Neighbour Discovery Protocol implemented in the TCP/IP stack shall provide support for manual configuration of Prefix List entries as described in IETF RFC 5942.
Rationale:	–
Use Case:	In-vehicle communication and diagnostics
Dependencies:	–
Supporting Material:	[19] IETF RFC 5942;

](RS_BRF_01784)

[SRS_Eth_00091] The Optimistic Duplicate Address Detection (DAD) for IPv6 shall be implemented according to IETF RFC 4429 [

Description:	The Optimistic Duplicate Address Detection (DAD) for IPv6 shall at least be implemented in the TCP/IP stack as stated in IETF RFC 4429.
Rationale:	–
Use Case:	In-vehicle communication and diagnostics
Dependencies:	[18] IETF RFC 4861
Supporting Material:	[20] IETF RFC 4429; [18] IETF RFC 4861

](RS_BRF_01784)

[SRS_Eth_00092] The IPv6 Addressing Architecture shall be implemented according to IETF RFC 4291 [

Description:	The IPv6 Addressing Architecture shall at least be implemented in the TCP/IP stack as stated in IETF RFC 4291. Limitation: Section 2.8 A Node's Required Addresses shall be limited to the node requirements for host only.
Rationale:	–
Use Case:	In-vehicle communication and diagnostics
Dependencies:	–
Supporting Material:	[21] IETF RFC 4291

](RS_BRF_01784)

[SRS_Eth_00093] The Transmission of IPv6 Packets shall be implemented according to IETF RFC 2464 [

Description:	The Transmission of IPv6 Packets over Ethernet Networks shall at least be implemented in the TCP/IP stack as stated in IETF RFC 2464.
---------------------	---



△

Rationale:	–
Use Case:	In-vehicle communication and diagnostics
Dependencies:	–
Supporting Material:	[22] IETF RFC 2464

](RS_BRF_01784)

[SRS_Eth_00094] The Default Address Selection for IPv6 shall be implemented according to IETF RFC 6724 [

Description:	The Default Address Selection for IPv6 shall at least be implemented in the TCP/IP stack as stated in IETF RFC 6724. Limitation: Only Section 5. Source Address Selection shall be supported
Rationale:	–
Use Case:	In-vehicle communication and diagnostics
Dependencies:	–
Supporting Material:	[23] IETF RFC 6724

](RS_BRF_01784)

[SRS_Eth_00095] The Handling of Overlapping IPv6 Fragments shall be implemented according to IETF RFC 5722 [

Description:	The Handling of Overlapping IPv6 Fragments shall at least be implemented in the TCP/IP stack as stated in IETF RFC 5722. Limitation: Only section 4. Node Behavior, first paragraph shall be supported.
Rationale:	For security reason, the overlapping of IP Fragments is explicitly forbidden (transmission and reception).
Use Case:	In-vehicle communication and diagnostics
Dependencies:	[15] IETF RFC 2460
Supporting Material:	[24] IETF RFC 5722

](RS_BRF_01784)

[SRS_Eth_00096] The Stateless Address Autoconfiguration for IPv6 shall be implemented according to IETF RFC 4862 [

Description:	The Stateless Address Autoconfiguration for IPv6 shall at least be implemented in the TCP/IP stack as stated in IETF RFC 4862. Limitation: Only the sections 5.1. Node Configuration Variables, 5.3. Creation of Link-Local Addresses, 5.4. Duplicate Address Detection, 5.5 Creation of Global Addresses and section 5.6 Configuration Consistency shall be supported.
Rationale:	–
Use Case:	In-vehicle communication and diagnostics
Dependencies:	–
Supporting Material:	[25] IETF RFC 4862

](RS_BRF_01784)

[SRS_Eth_00097] The Path MTU Discovery for IPv6 shall be implemented according to IETF RFC 1981 [

Description:	The Path MTU Discovery for IPv6 shall at least be implemented in the TCP/IP stack as stated in IETF RFC 1981.
Rationale:	If a packet exceeds the configured MTU size, the packet shall be discarded.
Use Case:	In-vehicle communication and diagnostics
Dependencies:	–
Supporting Material:	[17] IETF RFC 1981

](RS_BRF_01784)

4.3.1.5 TCP/IP ICMP

[SRS_Eth_00061] An API shall be available to generate any ICMP message [

Description:	An API Shall be available to generate any ICMP message
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	–

](RS_BRF_01784)

[SRS_Eth_00062] A call-back function shall be configurable for any received ICMP message type [

Description:	A call-back function shall be configurable for any received ICMP message type
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	–

](RS_BRF_01784)

4.3.1.6 TCP/IP Sub Module: ICMPv4

[SRS_Eth_00016] ICMPv4 shall be implemented according to IETF RFC 792 [

Description:	For the Internet Control Message Protocol (ICMP) shall at least be implemented in the TCP/IP stack as stated in IETF RFC 792
Use Case:	In-vehicle communication and diagnostics
Dependencies:	[SRS_Eth_00014]
Supporting Material:	[5] IETF RFC 792

](RS_BRF_01784)

4.3.1.7 TCP/IP Sub Module: ICMPv6

[SRS_Eth_00098] ICMPv6 shall be implemented according to IETF RFC 4443 [

Description:	Internet Control Message Protocol (ICMPv6) for IPv6 shall at least be implemented in the TCP/IP stack as stated in IETF RFC 4443
Rationale:	–
Use Case:	In-vehicle communication and diagnostics
Dependencies:	–
Supporting Material:	[8] IETF RFC 4443

](RS_BRF_01784)

4.3.1.8 TCP/IP TCP

[SRS_Eth_00017] TCP shall be implemented according to IETF RFC 793 [

Description:	The Transmission Control Protocol (TCP) shall at least be implemented in the TCP/IP stack as stated in IETF RFC 793
Use Case:	In-vehicle communication and diagnostics
Dependencies:	–
Supporting Material:	[6] IETF RFC 793

](RS_BRF_01784)

[SRS_Eth_00099] Congestion Control strategies shall be implemented according to IETF RFC 5681 [

Description:	The Congestion control strategies shall at least be implemented in the TCP/IP stack as stated in IETF RFC 5681. Limitation: The strategies Slow-Start, Congestion Avoidance, Fast Retransmit and Fast Recovery shall be implemented.
Rationale:	–
Use Case:	In-vehicle communication and diagnostics
Dependencies:	–
Supporting Material:	[26] IETF RFC 5681

](RS_BRF_01784)

[SRS_Eth_00100] The NewReno Modification shall be implemented according to IETF RFC 6582 [

Description:	The NewReno Modification to TCP's Fast Recovery Algorithm shall at least be implemented in the TCP/IP stack as stated in IETF RFC 6582. Limitation: The modification shall only be used if the Fast Recovery strategy of IETF RFC 5681 is enabled.
Rationale:	–
Use Case:	In-vehicle communication and diagnostics
Dependencies:	[26] IETF RFC 5681
Supporting Material:	[27] IETF RFC 6582

](RS_BRF_01784)

[SRS_Eth_00109] TCP shall support the Nagle algorithm according to IETF RFC 896 [

Description:	The "Congestion Control in IP/TCP Internetworks" aka Nagle algorithm shall be implemented in the TCP/IP stack as stated in IETF RFC 896.
Rationale:	–
Use Case:	In-vehicle communication and diagnostics
Dependencies:	–
Supporting Material:	[28] IETF RFC 896;

]([RS_BRF_01784](#))

4.3.1.9 TCP/IP TLS

[SRS_Eth_00133] TCP sockets shall be assigned to TLS connections [

Description:	It shall be configurable if a TCP socket uses TLS. The configuration also implies if the connection for this socket is for a TLS server or client.
Rationale:	Dedicated sockets shall use TLS.
Use Case:	A tester that communicates over a specific port number shall be forced to use TLS.
Dependencies:	–
Supporting Material:	–

]([RS_Main_01008](#), [RS_Main_00190](#), [RS_Main_00420](#))

[SRS_Eth_00134] Configuration of ciphersuites for TLS connections [

Description:	To limit the amount of crypto jobs that are necessary for TLS, AUTOSAR shall support the pre-configuration of a sub-set of ciphersuites.
Rationale:	Only those ciphersuites shall be offered for or by a dedicated TLS connection that are needed. This reduces the required resources within the ECU.
Use Case:	The TLS server and client offer a number of ciphersuites so that the server depicts one for best practice.
Dependencies:	–
Supporting Material:	–

]([RS_Main_00161](#))

[SRS_Eth_00135] The number of TLS connections that can be opened in parallel shall be configurable [

Description:	An ECU may support more than one secure port for TLS communication. To determine the number of TLS connections that can be active at the same time, the amount of TLS connections shall be configurable.
Rationale:	Minimize resource consumption. Not all possible communication links are used at the same time.
Use Case:	Support of several TLS connections in parallel, e.g. external and internal diagnostic tester
Dependencies:	–
Supporting Material:	–

](RS_Main_00161)

[SRS_Eth_00136] The size of a TLS fragment length shall be configurable [

Description:	Data transmission over TCP/TLS is done on block basis. According to IETF, the maximum size of such a block is 16kB. The size of such a block shall be configurable but meet the requirements of the communication partner.
Rationale:	Minimize resource consumption.
Use Case:	–
Dependencies:	–
Supporting Material:	–

](RS_Main_00161)

[SRS_Eth_00137] PSK Identity to PSK mapping shall be possible using custom software. [

Description:	When using TLS with PSK, for the mapping on server side from PSK identity to stored key, custom OEM software shall be usable. Additional configurations for mapping PSK Identity to key can be provided.
Rationale:	Offer custom ways of handling PSK identities.
Use Case:	–
Dependencies:	–
Supporting Material:	–

](RS_Main_00161)

[SRS_Eth_00138] TLS shall support at least basic requirements as defined in IETF RFC 5246 for version 1.2 or higher [

Description:	AUTOSAR shall support the standardized protocol for transport layer security for at least version 1.2. Subsets are possible as long as security aspects are not violated.
Rationale:	Support an IETF standard.
Use Case:	–
Dependencies:	–
Supporting Material:	[29] IETF RFC 5246

]([RS_Main_00190](#), [RS_Main_00510](#), [RS_Main_01008](#), [RS_Main_00260](#))

[SRS_Eth_00139] TLS shall support elliptic curve cryptography as defined in IETF RFC 4492 [

Description:	AUTOSAR shall support elliptic curve cryptography for transport layer security.
Rationale:	Support TLS with elliptic curve support.
Use Case:	–
Dependencies:	–
Supporting Material:	[30] IETF RFC 4492

]([RS_Main_00190](#), [RS_Main_00510](#), [RS_Main_01008](#), [RS_Main_00260](#))

[SRS_Eth_00140] TLS for diagnostic communication (DoIP) shall support at least one ciphersuite as defined in ISO13400-2. [

Description:	The DoIP specification ISO13400-2 defines a list of ciphersuites. If a TLS connection is used for diagnostic communication over TLS, at least one of the defined ciphersuites shall be supported.
Rationale:	Support ISO 13400-2
Use Case:	–
Dependencies:	[4] ISO13400-2
Supporting Material:	–

]([RS_Main_00190](#), [RS_Main_00510](#), [RS_Main_01008](#), [RS_Main_00260](#))

[SRS_Eth_00141] TLS shall support the use of pre-shared keys as defined in IETF RFC 4279 [

Description:	AUTOSAR shall support the standardized protocol for transport layer security using pre-shared keys. Subsets are possible as long as security aspects are not violated.
Rationale:	Support an IETF standard.





Use Case:	–
Dependencies:	–
Supporting Material:	[31] IETF RFC 4279

]([RS_Main_00190](#), [RS_Main_00510](#), [RS_Main_01008](#), [RS_Main_00260](#))

4.3.1.10 TCP/IP UDP

[SRS_Eth_00018] UDP shall be implemented according to IETF RFC 768 [

Description:	The User Datagram Protocol (UDP) shall at least be implemented in the TCP/IP stack as stated in IETF RFC 768
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	[32] IETF RFC 768

]([RS_BRF_01784](#))

4.3.1.11 TCP/IP DHCP

[SRS_Eth_00065] An API shall be available to fill DHCP options field [

Description:	An API shall be available to fill any DHCPv4 and DHCPv6 options field
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01784](#))

[SRS_Eth_00066] An API shall be available to read any received DHCP options field [

Description:	An API shall be available to read any received DHCPv4 and DHCPv6 options field
Rationale:	–
Use Case:	–





Dependencies:	–
Supporting Material:	–

](RS_BRF_01784)

[SRS_Eth_00088] DHCP Server [

Description:	IP-addresses can be assigned depending on the port a DHCP request has been received.
Rationale:	DHCP server functionality is a standard feature in IP-networks and shall be supported. In automotive networks a modification is necessary that allows to assign defined IP addresses on a port basis.
Use Case:	ECU without a predefined IP address can get an IP address assigned via DHCP.
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776, RS_BRF_01784)

4.3.1.12 TCP/IP Sub-Module: DHCPv4

[SRS_Eth_00020] DHCPv4 shall be implemented according to IETF RFC 2131 [

Description:	The Dynamic Host Configuration Protocol (DHCP) shall at least be implemented in the TCP/IP stack as stated in IETF RFC 2131
Rationale:	–
Use Case:	–
Supporting Material:	[33] IETF RFC 2131

](RS_BRF_01784)

[SRS_Eth_00021] The DHCPv4 host name option shall be implemented according to IETF RFC 2132 [

Description:	The DHCPv4 host name option shall at least be implemented in the TCP/IP stack as stated in IETF RFC 2132
Rationale:	–
Use Case:	–
Dependencies:	[SRS_Eth_00047]; SRS_Eth_00020





Supporting Material:	[34] IETF RFC 2132
-----------------------------	--------------------

](RS_BRF_01784)

[SRS_Eth_00101] The Fully Qualified Domain Name (FQDN) Option for DHCPv4 shall be implemented according to IETF RFC 4702 [

Description:	The Fully Qualified Domain Name Option for Dynamic Host Configuration Protocol for IPv4 shall at least be implemented in the TCP/IP stack as stated in IETF RFC 4702 (The Dynamic Host Configuration Protocol for IPv4 (DHCPv4) Client Fully Qualified Domain Name (FQDN) Option).
Rationale:	–
Use Case:	–
Supporting Material:	[35] IETF RFC 4702

](RS_BRF_01784)

4.3.1.13 TCP/IP Sub-Module: DHCPv6

[SRS_Eth_00067] The IPv6 stack shall implement DHCPv6 as a DHCPv6 client according to IETF RFC 3315 [

Description:	<p>The Dynamic Host Configuration Protocol for IPv6 (DHCPv6) shall at least be implemented in the TCP/IP stack as stated in IETF RFC 3315 as a DHCPv6 client.</p> <p>Limitation: The following sections shall not be implemented according to the "only Client" Use Case:</p> <ul style="list-style-type: none"> • Relay Agent Behavior • Server Behavior • Section 12. Management of Temporary Addresses • Section 21. Authentication of DHCP Messages • Section 22.5. Identity Association for Temporary Addresses Option • Section 22.11. Authentication Option • Section 22.14. Rapid Commit Option
Rationale:	–
Use Case:	In-vehicle communication and diagnostics
Dependencies:	–
Supporting Material:	[36] IETF RFC 3315

](RS_BRF_01784)

[SRS_Eth_00102] The Client Fully Qualified Domain Name (FQDN) Option for IPv6 shall be implemented according to IETF RFC 4704 [

Description:	The Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Client Fully Qualified Domain Name (FQDN) Option shall at least be implemented in the TCP/IP stack as stated in IETF RFC 4704. Limitation: Only the client behavior shall be supported
Rationale:	–
Use Case:	In-vehicle communication and diagnostics
Dependencies:	[36] IETF RFC 3315
Supporting Material:	[37] IETF RFC 4704

] ([RS_BRF_01784](#))

[SRS_Eth_00068] The DNS configuration options for DHCPv6 shall be supported by IETF RFC 3646 [

Description:	The DNS Configuration Options for Dynamic Host Configuration Protocol for IPv6 shall be supported by IETF RFC 3646.
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	[38] IETF RFC 3646; [36] IETF RFC 3315

] ([RS_BRF_01784](#))

4.3.1.14 TCP/IP IPsec

[SRS_Eth_00142] The Security Architecture for the Internet Protocol shall be implemented according to IETF RFC 4301 [

Description:	The Security Architecture for the Internet Protocol shall be implemented in the TCP/IP stack as stated in IETF RFC 4301. Limitation: Any sections related to tunnel mode, may or may not be implemented, e.g. section 5.1.2, 7.1, 7.2
Rationale:	Ethernet is mostly used to connect to other AUTOSAR bus nodes. They use these protocols; therefore, they need to be available inside AUTOSAR
Use Case:	In-vehicle secure communication
Dependencies:	–
Supporting Material:	[39] IETF RFC 4301

] ([RS_Main_00280](#), [RS_Main_00430](#))

[SRS_Eth_00143] The IP Authentication Header (AH) shall be implemented according to IETF RFC 4302 [

Description:	The IP Authentication Header (AH) shall be implemented in the TCP/IP stack as stated in IETF RFC 4302. Limitation: Section 3.1.2, related to tunnel mode, may or may not be implemented
Rationale:	Ethernet is mostly used to connect to other AUTOSAR bus nodes. They use these protocols; therefore, they need to be available inside AUTOSAR
Use Case:	In-vehicle secure communication
Dependencies:	–
Supporting Material:	[40] IETF RFC 4302

]([RS_Main_00280](#), [RS_Main_00430](#))

[SRS_Eth_00144]{DRAFT} IP Encapsulating Security Payload (ESP) shall be implemented according to IETF RFC 4303 [

Description:	The IP Encapsulating Security Payload (ESP) shall be implemented in the TCP/IP stack as stated in IETF RFC 4303. Limitation: Any section related to tunnel mode, may or may not be implemented, e.g. section 3.1.2
Rationale:	Ethernet is mostly used to connect to other AUTOSAR bus nodes. They use these protocols; therefore, they need to be available inside AUTOSAR
Use Case:	In-vehicle secure communication
Dependencies:	–
Supporting Material:	[41] IETF RFC 4303

]([RS_Main_00280](#), [RS_Main_00430](#))

[SRS_Eth_00145] The Internet Key Exchange (IKEv2) Protocol shall be implemented according to IETF RFC 7296 [

Description:	The Security Architecture for the Internet Protocol shall be implemented in the TCP/IP stack as stated in IETF RFC 7296. Limitation: Support is limited to scenario 1.1.2 'Endpoint-to-Endpoint Transport'
Rationale:	Ethernet is mostly used to connect to other AUTOSAR bus nodes. They use these protocols; therefore, they need to be available inside AUTOSAR
Use Case:	In-vehicle secure communication
Dependencies:	–
Supporting Material:	[42] IETF RFC 7296

]([RS_Main_00280](#), [RS_Main_00430](#))

4.4 SWS Socket Adaptor (SoAd)

[SRS_Eth_00004] The SoAd shall support a local multi-homed host [

Description:	The SoAd shall support a local multi-homed host.
Rationale:	An ECU might be connected to multiple IP networks for different use-cases
Use Case:	–
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

[SRS_Eth_00002] The IP addresses as well as the method of acquisition shall be a configurable item. [

Description:	The IP addresses as well as the method of acquisition shall be a configurable item.
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

[SRS_Eth_00001] The initialization the SoAd shall be able to establish all TCP connections [

Description:	The initialization the SoAd shall be able to establish all TCP connections as described in the configuration.
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

[SRS_Eth_00005] Both UDP or TCP shall be usable [

Description:	Both UDP or TCP shall be usable.
Rationale:	–
Use Case:	–
Dependencies:	–



△

Supporting Material:	–
-----------------------------	---

](RS_BRF_01776)

[SRS_Eth_00009] Upon Shutdown the Socket Adaptor shall close all open TCP connections [

Description:	Upon Shutdown the Socket Adaptor shall close all open TCP connections.
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00008] The Socket Adaptor shall immediately try to re-establish any TCP connection if it is lost [

Description:	The Socket Adaptor shall immediately try to re-establish any TCP connection if it is lost. 'never give up' strategy
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00011] The resources required for the socket connections shall be predictable by analyzing the configuration information [

Description:	The resources required for the socket connections shall be predictable by analyzing the configuration information.
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00012] An API shall be offered where buffer memory is required [

Description:	The Socket Adaptor, Tcplp, Ethlf and Eth shall offer an API where buffer memory for transmission and reception of AUTOSAR PDUs via Ethernet is required
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	AUTOSAR API will be able to satisfy this requirement!

](RS_BRF_01776)

[SRS_Eth_00013] An API shall be offered for reduced copy operation [

Description:	The Socket Adaptor, Tcplp, Ethlf and Eth shall offer an API where the number of copy operations for transmission and reception of AUTOSAR PDUs via Ethernet is optimized (i.e. reduced to a minimum)
Rationale:	–
Use Case:	AUTOSAR API
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00006] The Socket Adaptor shall be able to transfer data using TCP/IP without the introduction of additional protocol overhead [

Description:	The Socket Adaptor shall be able to transfer data using TCP/IP without the introduction of additional protocol overhead.
Rationale:	The protocol overhead introduced by Ethernet and TCP/IP is so large, that additional overhead is considered harmful.
Use Case:	–
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00048] SoAd shall implement a mechanism to bi-directionally route PDUs between an AUTOSAR connector and the TCP/IP stack [

Description:	SoAd shall implement a mechanism to bi-directionally route PDUs between an AUTOSAR connector and the TCP/IP stack.
Rationale:	This is necessary to allow communication
Use Case:	–



△

Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00049] SoAd shall implement an API towards higher layers, which is equivalent to the API provided by interface modules [

Description:	SoAd shall implement an API towards higher layers, which is equivalent to the API provided by interface modules like FrIf, CanIf and LinIf.
Rationale:	Higher layers shall not be aware of the underlining communication stack.
Use Case:	–
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00050] SoAd shall implement an API towards higher layers, which is equivalent to the API provided by transport layer [

Description:	SoAd shall implement an API towards higher layers, which is equivalent to the API provided by transport layer modules like FrTP and CanTP.
Rationale:	Higher layers shall not be aware of the underlining communication stack.
Use Case:	–
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00051] The Socket Adaptor shall specify and be able to use a call-back interface towards the TCP/IP stack [

Description:	The Socket Adaptor shall specify and be able to use a call-back interface towards the TCP/IP stack.
Rationale:	the SoAd shall specify and be able to make use of a call-back interface, that reduces the number of required copy actions to and from the TCP/IP stack (optimized parameter) and reduces the latency (call-back functions).
Use Case:	Instead of a commercial of the shelf TCP/IP stack an optimized automotive TCP/IP stack shall be usable.
Dependencies:	–

](RS_BRF_01776)

[SRS_Eth_00069] The Socket Adaptor shall implement a mechanism to share multiple PDUs from/to the same or different upper modules [

Description:	The SoAd shall implement a mechanism by which a socket connection can be shared among multiple PDUs from/to the same or different upper layer modules.
Rationale:	Minimize resource consumption
Use Case:	Transmission of multiple COM-PDUs to the same remote node.
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

[SRS_Eth_00070] The Socket Adaptor shall implement a mechanism to transmit PDUs to more than one receiver [

Description:	The SoAd shall implement a mechanism by which a PDU shall be transmitted to more than one receiver using the same or a different socket connection and by which a received PDU shall be forwarded to more than one upper layer PDU using the same or a different socket connection to the same or a different upper layer module. (i.e. fan out to lower and upper layer, fan in from multiple connections)
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

[SRS_Eth_00116] The Socket Adaptor shall implement a mechanism to transmit multiple PDUs within the same UDP datagram [

Description:	The SoAd shall implement a mechanism by which it shall be possible to transmit multiple PDUs within the same UDP datagram.
Rationale:	Improved bandwidth utilization
Use Case:	Efficient data transmission
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#), [RS_BRF_01716](#))

[SRS_Eth_00071] The Socket Adaptor shall implement a mechanism to activate or deactivate an upper layer using a routing group [

Description:	The SoAd shall implement a mechanism by which a routing group can be activated or deactivated by an upper layer during runtime. A routing group consist of a number of PDUs from the same or different socket connections which shall be blocked (i.e. not routed to or from the upper layer). A PDU may only be part of a single routing group.
Rationale:	Blocking of undesired messages
Use Case:	Blocking of undesired notifications from a Publish/Subscribe Server
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00124] The SoAd shall implement mechanisms to share the same PDU pair for the reception from and transmission to multiple remote nodes [

Description:	The SoAd shall support configurations that relate multiple remote nodes to one single receive PDU for receptions as well as to one single transmit PDU for transmissions. Upon reception of such a PDU, the SoAd shall be able to provide additional information regarding the remote node to the upper layer. Upon transmission the SoAd shall be able to select one of the multiple possible targets based on additional information provided by the upper layer.
Rationale:	This allows to deliver the response data to the same remote node that requested it, without explicit configuration of all possible data paths.
Use Case:	Scalability of Client/Server Communication
Dependencies:	–
Supporting Material:	–

](RS_BRF_01317)

[SRS_Eth_00131] The SoAd shall support access to measurement counter values [

Description:	The SoAd shall provide an optional API to access measurement counter values, like for example the number of dropped PDUs
Rationale:	–
Use Case:	Interface for diagnostics
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

4.5 SWS Diagnostics over IP (DoIP)

[SRS_Eth_00047] DoIP shall be able to access the DHCP host name option. [

Description:	The value used in the DHCP host name option shall be accessible and changeable by DoIP.
Rationale:	–
Use Case:	–
Dependencies:	[SRS_Eth_00021]
Supporting Material:	[4] ISO13400-2

](RS_BRF_01440)

[SRS_Eth_00024] DoIP messages shall be bi-directionally routed [

Description:	DoIP shall implement a mechanism to bi-directionally route DoIP messages between the vehicle networks and TCP sockets.
Rationale:	This is necessary to allow DoIP communication
Use Case:	–
Dependencies:	–
Supporting Material:	[4] ISO13400-2

](RS_BRF_01440)

[SRS_Eth_00025] Valid DoIP messages shall be recognized [

Description:	DoIP shall provide a mechanism to recognize valid DoIP Messages
Rationale:	This is necessary to ensure correct DoIP communication
Use Case:	–
Dependencies:	–
Supporting Material:	[4] ISO13400-2

](RS_BRF_01440)

[SRS_Eth_00026] DoIP Vehicle Identification shall be provided [

Description:	DoIP shall provide a mechanism to identify a vehicle respective its DoIP entity in a network
Rationale:	This is necessary to determine the IP addresses of the DoIP entity
Use Case:	–
Dependencies:	–



△

Supporting Material:	[4] ISO13400-2
-----------------------------	----------------

](RS_BRF_01440)

[SRS_Eth_00027] DoIP diagnostic message shall have a format [

Description:	DoIP shall implement a message format to allow the routing of diagnostic messages
Rationale:	This is necessary to allow DoIP
Use Case:	–
Dependencies:	–
Supporting Material:	[4] ISO13400-2

](RS_BRF_01440)

[SRS_Eth_00028] Multiple DoIP sockets shall be allowed on a single port [

Description:	DoIP shall implement a mechanism to allow the use and assignment of multiple sockets on a single port, while ensuring that no active communication is disturbed
Rationale:	This is necessary for the efficient use of socket communication
Use Case:	–
Dependencies:	–
Supporting Material:	[4] ISO13400-2

](RS_BRF_01440)

[SRS_Eth_00080] DoIP shall implement a mechanism to retrieve diagnostic power mode [

Description:	DoIP shall implement a mechanism to retrieve the diagnostic power mode of a DoIP Entity via a diagnostic power mode request as described in ISO 13400-2:2012. The service is needed to provide the tester with the diagnostic capability information.
Rationale:	This is necessary to allow DoIP
Use Case:	–
Dependencies:	–
Supporting Material:	[4] ISO13400-2

](RS_BRF_01440)

[SRS_Eth_00081] DoIP shall be able to dynamically maintain connection to different testers [

Description:	DoIP shall be able to dynamically maintain the connection to different testers as described in ISO 13400-2:2012
Rationale:	This is necessary to allow DoIP
Use Case:	–
Dependencies:	–
Supporting Material:	[4] ISO13400-2

]([RS_BRF_01440](#))

[SRS_Eth_00082] DoIP shall implement a mechanism to retrieve Entity Status [

Description:	DoIP shall implement a mechanism to retrieve the entity status information of a DoIP Entity via an Entity status request as described in ISO 13400-2:2012. The service is needed to provide the tester with the diagnostic capability information
Rationale:	This is necessary to allow DoIP
Use Case:	–
Dependencies:	–
Supporting Material:	[4] ISO13400-2

]([RS_BRF_01440](#))

[SRS_Eth_00083] DoIP shall implement a mechanism to check if diagnostic testers are alive [

Description:	DoIP shall implement a mechanism to check if the connected testers are still available to allow an efficient management of connections for diagnostics from the DoIP entity point of view. The alive check mechanism shall be according to 13400-2:2012.
Rationale:	This is necessary to allow DoIP
Use Case:	–
Dependencies:	–
Supporting Material:	[4] ISO13400-2

]([RS_BRF_01440](#))

[SRS_Eth_00084] DoIP shall implement routing activation mechanism [

Description:	DoIP shall implement a mechanism to allow the selective activation of diagnostics on all supported socket connections with the Routing activation mechanisms described in ISO 13400-2:2012
---------------------	--



△

Rationale:	This is necessary to allow DoIP
Use Case:	–
Dependencies:	–
Supporting Material:	[4] ISO13400-2

](RS_BRF_01440)

[SRS_Eth_00104] DoIP shall support USDT and UUDT messages according to ISO 14229-5 [

Description:	DoIP module shall support both the interface-API (for UUDT) and the transport layer API towards the PduR for USDT.
Rationale:	DoIP shall be able to route both USDT (Unacknowledged segmented data transfer) and UUDT (Unacknowledged unsegmented data transfer) messages.
Use Case:	Routing of USDT and UUDT messages
Dependencies:	–
Supporting Material:	[43] ISO 14229-5

](RS_BRF_01440)

4.6 SWS Ethernet Interface (EthIf)

[SRS_Eth_00031] The Ethernet Interface shall be pre-compile time configurable for interrupt or polling. [

Description:	The Ethernet Interface shall provide configuration for interrupt and polling mode. In interrupt mode received frames shall be reported from the driver. In polling mode the main function shall check for received frames.
Rationale:	Interface and interaction
Use Case:	For low latency interrupt mode is crucial. On systems with high system load the polling mode reduces the system load
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00032] The Ethernet Interface shall provide hardware configuration and initialization. [

Description:	The Ethernet Interface shall provide hardware independent configuration and initialization interface.
Rationale:	Hardware abstraction
Use Case:	Exchanging the used Ethernet controller and transceiver shall be transparent and only be reflected by replacing the used driver.
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

[SRS_Eth_00033] The Ethernet Interface shall provide indication for link state change. [

Description:	The Ethernet Interface shall provide indication for link state change of connected transceivers.
Rationale:	Interface and interaction
Use Case:	Disconnection of the cable results in invalid IP address. Thus a valid IP assignment cycle has to be started.
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

[SRS_Eth_00072] The Ethernet Interface shall provide VLAN support [

Description:	The Ethernet Interface shall support VLAN tagging of Ethernet frames according to IEEE 802.1Q, i.e. add/remove VLAN tag at transmission/reception of Ethernet frames. A VLAN shall appear as a separate EthIf controller to the upper layer. The VLAN identifier shall be specified via EthIf configuration; the VLAN priority shall be specified by the upper layer as part of the transmission request and ignored at reception.
Rationale:	Separation of logical networks
Use Case:	Separation of internal and external vehicle communication
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

[SRS_Eth_00132] The Ethernet Interface shall support access to measurement counter values [

Description:	The Ethernet Interface shall provide an optional API to access measurement counter values, like for example the number of dropped frames
Rationale:	–
Use Case:	Interface for diagnostics
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00156]{DRAFT} The Ethernet Interface shall provide indication for a received sleep request. [

Description:	OPEN ALLIANCE (OA) TC10 Sleep/Wake-up specification for Automotive Ethernet define the sleep and wake-up capability over data line. The EthIf shall forward a received sleep request to EthSM to shut down the communication channel if it's configured as a communication slave.
Rationale:	Interface and interaction
Use Case:	If an ECU acts as an communication slave on the communication channel, the ECU shall follow the communication requests of its communication master
Supporting Material:	[44] OPEN ALLIANCE Sleep/Wake-up Specification Version 2.0 (Rel Feb 21, 2017)

](RS_Main_00430)

[SRS_Eth_00157] The Ethernet Interface shall trigger requested modes for Ethernet hardware with wake-up capability even if the requested mode has already been reached. [

Description:	The Ethernet Interface shall re-trigger requested modes for Ethernet hardware with wake-up capability even if the requested mode has already been reached
Rationale:	An Ethernet switch network need to trigger wake-up and propagate the wake-up on the whole network to signal connected ECUs about a communication request
Use Case:	Wake-up of the Ethernet switched network, due to a PNC request.
Supporting Material:	[44] OPEN ALLIANCE Sleep/Wake-up Specification Version 2.0 (Rel Feb 21, 2017)

](RS_Main_00430)

4.7 SWS Ethernet Driver (Eth)

[SRS_Eth_00035] The Ethernet Driver shall be pre-compile time configurable for interrupt or polling. [

Description:	The Ethernet Driver shall provide configuration for interrupt and polling mode. In interrupt mode received frames shall be reported from the driver.
Use Case:	For low latency interrupt mode is crucial. On systems with high system load the polling mode reduces the system load
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

[SRS_Eth_00036] The Ethernet Driver shall provide hardware configuration and initialization. [

Description:	The Ethernet Driver shall provide hardware independent configuration and initialization interface.
Rationale:	Hardware abstraction
Use Case:	Exchanging the used Ethernet controller shall be transparent and only be reflected by replacing the used driver.
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

[SRS_Eth_00073] The Ethernet Driver shall support to receive MAC broadcast frames [

Description:	The Ethernet driver shall support the configuration of the Ethernet controller to receive MAC broadcast frames.
Rationale:	Reception of frames by all nodes
Use Case:	ARP Request
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

[SRS_Eth_00074] The Ethernet Driver shall support to receive MAC groupcast frames [

Description:	The Ethernet driver shall support the configuration of the Ethernet controller to receive MAC multicast frames.
Rationale:	Reception of frames by multiple nodes
Use Case:	Notification messages





Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00105] Support of time stamping in hardware [

Description:	The Ethernet driver shall provide an implementation which allows time stamping of Ethernet frames for time synchronized messages. This implementation has to be compatible as defined in "[9], ANNEX B.1.2 Time measurement granularity". If this granularity could be ensured less than 100% by the supported hardware an additional software solution within the driver might be needed. However, a 100% software solution, by using resources outside of the Ethernet controller, e.g. own GPT etc., is not recommended.
Rationale:	Efficient implementation without hidden hardware resources
Use Case:	GlobalTimeSynchronization
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00127] The Ethernet Driver shall provide statistic counter values [

Description:	The Ethernet Driver shall provide interfaces to read out statistic counter values (such as number of received, transmitted and dropped frames, etc.) for diagnostic analysis.
Rationale:	–
Use Case:	Diagnostics
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00146] The Ethernet Driver shall provide 10BASE-T1S support [

Description:	The Ethernet Driver shall support 10BASE-T1S compliant hardware
Rationale:	Hardware abstraction
Use Case:	Support of ECUs which are connected via 10BASE-T1S compliant hardware
Dependencies:	–





Supporting Material:	[45] OPEN Alliance TC14
-----------------------------	-------------------------

](RS_BRF_01776)

[SRS_Eth_00147]{DRAFT} The Ethernet Driver shall support SPI [

Description:	The Ethernet Driver shall support SPI interface
Rationale:	Hardware access
Use Case:	Support of ECUs which are using 10BASE-T1S compliant hardware
Dependencies:	–
Supporting Material:	[45] OPEN Alliance TC14

](RS_BRF_01776)

[SRS_Eth_00148] The Ethernet Driver shall support MII [

Description:	The Ethernet Driver shall support MII interface
Rationale:	Hardware access
Use Case:	Support of ECUs which are using 10BASE-T1S compliant hardware
Dependencies:	–
Supporting Material:	[45] OPEN Alliance TC14

](RS_BRF_01776)

4.8 SWS Ethernet Transceiver Driver (EthTrcv)

[SRS_Eth_00039] The Ethernet Transceiver Driver shall provide hardware configuration and initialization. [

Description:	The Ethernet Transceiver Driver shall provide hardware independent configuration and initialization interface.
Rationale:	Hardware abstraction
Use Case:	Exchanging the used Ethernet transceiver shall be transparent and only be reflected by replacing the used driver.
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00040] The Ethernet Transceiver Driver shall provide access to the link state. [

Description:	The Ethernet Transceiver Driver shall provide access to the link state of connected transceivers.
Rationale:	Interface and interaction
Use Case:	Disconnection of the cable results in invalid IP address. Thus a valid IP assignment cycle has to be started.
Dependencies:	–
Supporting Material:	–

] ([RS_BRF_01776](#))

[SRS_Eth_00117] The Ethernet Transceiver Driver shall provide access to standardized hardware features [

Description:	The Ethernet Transceiver Driver shall provide access to standardized hardware features such as signal quality measurement and physical layer test modes.
Rationale:	Hardware abstraction
Use Case:	Diagnostics and monitoring
Dependencies:	–
Supporting Material:	–

] ([RS_BRF_01776](#), [RS_BRF_01920](#))

[SRS_Eth_00106] The Ethernet Transceiver Driver shall switch on/off wake up functionality at pre compile time. [

Description:	The Ethernet Transceiver Driver shall enable and disable wake up functionality of connected transceivers if supported by the hardware.
Rationale:	Support the transceivers' wake-up capabilities
Use Case:	Wake up the ECU by bus
Dependencies:	–
Supporting Material:	–

] ([RS_BRF_01104](#), [RS_BRF_01776](#))

[SRS_Eth_00107] The Ethernet Transceiver Driver shall support access to the wake up reason. [

Description:	The Ethernet Transceiver Driver shall provide access to the wake up reason of connected transceivers if supported by the hardware.
Rationale:	Distinguish between wake-up by bus or other wake-up
Use Case:	Different handling of the wake-up event (e.g. starting/non starting of communication after a wake-up)
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01104](#), [RS_BRF_01776](#))

[SRS_Eth_00108] The Ethernet Transceiver Driver shall be able to wake-up an Ethernet network. [

Description:	The Ethernet stack shall be able to wake-up an Ethernet network if supported by hardware.
Rationale:	Support the Ethernet PHY's wake-up capabilities
Use Case:	Wake up the Ethernet network actively on dataline
Dependencies:	–
Supporting Material:	[44] OPEN ALLIANCE Sleep/Wake-up Specification Version 2.0 (Rel Feb 21, 2017)

]([RS_BRF_01104](#), [RS_BRF_01776](#), [RS_Main_00430](#))

[SRS_Eth_00151] The Ethernet Transceiver Driver shall support a controlled link shutdown (sleep request) [

Description:	OPEN ALLIANCE (OA) TC10 Sleep/Wake-up specification for Automotive Ethernet define the sleep capability over data line. Therefore the EthTrcv shall provide a possibility to trigger a sleep request to shut down the link with a confirmation of the linked communication partner.
Rationale:	OA TC10 compatible hardware provide sleep handling to put linked communication partner synchronous in a sleep mode
Use Case:	Used to set the Ethernet hardware in a sleep mode after network management has finished the synchronous shut down of the communication.
Supporting Material:	[44] OPEN ALLIANCE Sleep/Wake-up Specification Version 2.0 (Rel Feb 21, 2017)

]([RS_Main_00430](#))

[SRS_Eth_00152]{DRAFT} The Ethernet Transceiver Driver shall be able to react on sleep request received from the network. [

Description:	OPEN ALLIANCE (OA) TC10 Sleep/Wake-up specification for Automotive Ethernet define the sleep and wake-up capability over data line. Therefore the EthTrcv shall provide a possibility to react on sleep request received from the link partner.
Rationale:	OA TC10 compatible hardware provide sleep handling to put linked communication partner synchronous in a sleep mode
Use Case:	If an ECU acts as a communication slave on the communication channel, the ECU shall follow the communication requests of its communication master
Supporting Material:	[44] OPEN ALLIANCE Sleep/Wake-up Specification Version 2.0 (Rel Feb 21, 2017)

](RS_Main_00430)

[SRS_Eth_00153] The Ethernet Transceiver Driver shall be able to configure the Ethernet hardware (PHY) according to the wake-up handling. [

Description:	OPEN ALLIANCE(OA) TC10 Sleep/Wake-up specification for Automotive Ethernet define the sleep and wake-up capability over data line. Therefore the Ethernet Transceiver Driver shall support a possibility to configure the Ethernet hardware regarding the wake-up handling: <ul style="list-style-type: none"> • Forward a remote received wake-up request • Forward a local received wake-up request • Enable / Disable wake-up request forwarding • Timing constraints regarding local wake-up request
Rationale:	OA TC10 compatible hardware provide wake-up forwarding to ensure a fast propagation of a wake-up request within an Ethernet switched network
Use Case:	An Ethernet switched network has to be configured regarding the wake-up forwarding to ensure a fast propagation of a wake-up request
Supporting Material:	[44] OPEN ALLIANCE Sleep/Wake-up Specification Version 2.0 (Rel Feb 21, 2017)

](RS_Main_00430)

[SRS_Eth_00154] The Ethernet Transceiver Driver shall provide a possibility to re-trigger a wake-up request. [

Description:	The Ethernet Transceiver Driver shall provide a possibility to re-trigger a wake-up request of active communication request
Rationale:	An Ethernet switched network need to re-trigger a wake-up request to ensure reliability and availability of communication requests within an Ethernet switched network. Repeating a wake-up request could overcome error scenarios where the propagation of a wake-up request was inhibited due to disturbing pulse on the data line.
Use Case:	Overcome error scenarios where disturbing pulse inhibit propagation of wake-up on data line





Supporting Material:	[44] OPEN ALLIANCE Sleep/Wake-up Specification Version 2.0 (Rel Feb 21, 2017)
-----------------------------	---

](RS_Main_00430)

[SRS_Eth_00155] The Ethernet Transceiver Driver shall provide a possibility to re-trigger a sleep request. [

Description:	The Ethernet Transceiver Driver shall provide a possibility to re-trigger a sleep request, if the sleep request was not accepted by the Ethernet hardware of the connected ECU in an Ethernet switched network
Rationale:	An ECU which is connected to an Ethernet switched network that uses OA TC10 compliant Ethernet hardware need to re-trigger a sleep request, if the Ethernet hardware of the connected ECU has not confirmed to transit to sleep mode. The ECU which initiated the sleep request, shall repeat the sleep request a configurable amount of time. If the repetitions of the sleep requests were not confirmed by the Ethernet hardware of the connected ECU, the ECU which initiate the sleep request shall shutdown the affected communication channel. This shall avoid to keep an ECU awake if the Ethernet hardware do not confirm a sleep request for some reasons (error scenario)
Use Case:	Overcome error scenarios where a sleep request
Supporting Material:	[44] OPEN ALLIANCE Sleep/Wake-up Specification Version 2.0 (Rel Feb 21, 2017)

](RS_Main_00430)

[SRS_Eth_00149] The Ethernet Transceiver Driver shall provide 10BASE-T1S support [

Description:	The Ethernet Transceiver Driver shall support 10BASE-T1S compliant hardware.
Rationale:	Hardware abstraction
Use Case:	Support of ECUs which are connected via 10BASE-T1S compliant hardware
Dependencies:	–
Supporting Material:	[45] OPEN Alliance TC14

](RS_BRF_01104, RS_BRF_01776)

[SRS_Eth_00150] The Ethernet Transceiver Driver shall provide MII interface [

Description:	The Ethernet Driver shall support MII interface
Rationale:	Hardware abstraction
Use Case:	Support of ECUs which are using 10BASE-T1S compliant hardware (standalone transceiver)





Dependencies:	–
Supporting Material:	[45] OPEN Alliance TC14

]([RS_BRF_01104](#), [RS_BRF_01776](#))

4.9 SWS Ethernet Switch Driver (EthSwT)

[SRS_Eth_00118] Transparent interface to underlying EthTrcv module(s) [

Description:	Ethernet Switch Driver shall provide a transparent interface to underlying Eth Trcv module(s) for transceiver related APIs (GetLinkState, GetBaudRate, Get DuplexMode, etc.)
Rationale:	Transceiver related interfaces like GetLinkState, GetBaudRate or GetDuplex Mode shall be accessible without the dedicated knowledge of the underlying hardware architecture of Switches and Transceivers.
Use Case:	Access to Transceiver Driver
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

[SRS_Eth_00119] Access to hardware status of ports [

Description:	The Ethernet Switch Driver shall provide the hardware status to the upper layer.
Rationale:	To read for example the link state the Ethernet Switch Driver needs to provide interfaces to read the status of the connected port hardware.
Use Case:	Access to ports
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

[SRS_Eth_00120] Hardware access via MII and/or SPI [

Description:	Ethernet Switch Driver shall provide an interface for managing and controlling over SPI and/or MII.
Rationale:	Automotive Ethernet networks can include managed switches with MII and/or SPI interfaces. The Ethernet Switch Driver shall give access to the needed interfaces. For the management, i.e. control and configuration of these switches and their ports, such an interface has to be provided.





Use Case:	Control Switch hardware with MII or SPI interface
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00121] Configuration of forwarding rules [

Description:	Ethernet Switch Driver shall provide an interface to configure forwarding rules of the hardware.
Rationale:	Ethernet Switch Driver shall provide an interface to configure forwarding rules like ARL tables, VLAN memberships, priority regenerations and transmit schedulers and shapers.
Use Case:	Control forwarding rules
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00122] Persistent storage of configurations [

Description:	Ethernet Switch Driver shall be able to store configurations persistently.
Rationale:	It shall be possible to store the Switch configuration in the non volatile memory.
Use Case:	Store Configuration persistently
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00087] Semi-Static Auto-Configuration [

Description:	Beside the possibility to statically configure a switch, switches are able to automatically learn certain parameters. This learning process should be executed at assembly time or after replacement of ECUs with spare parts. Therefore, a process needs to be defined which triggers the learning process in these scenarios and also allows a persistent storage of learned parameters.
Rationale:	This Semi-static Auto-Configuration will be used in assembly and maintenance situations where Ethernet components are connected for the first time or replaced by spare parts.
Use Case:	Necessary for each Ethernet-ECU
Dependencies:	–





Supporting Material:	–
-----------------------------	---

](RS_BRF_01776)

[SRS_Eth_00114] Ethernet Switch Filtering and Policing [

Description:	Ethernet Switch Driver shall support all necessary configuration parameters to allow filtering and policing if supported by the hardware.
Rationale:	These mechanisms can be used to avoid bursts, restrict babbling idiots and to prevent denial of service attacks.
Use Case:	Ensure bandwidth for critical applications
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00123] Testing and diagnostic of switch ports [

Description:	The Ethernet Switch Driver shall provide interfaces to support testing and diagnostic functionalities like port mirroring, cable diagnostics and port monitoring depending on the possibilities of the used hardware.
Rationale:	To use diagnostic functionalities of switches and the connected transceivers, the Ethernet Switch Driver shall provide interfaces to configure test modes and read diagnostic data from the switch and/or transceiver hardware.
Use Case:	Testing and Diagnostics
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00125] The Ethernet Switch Driver shall support switch frame management [

Description:	The Ethernet Switch Driver shall enable switch frame management control for Ethernet frames regarding a switch port specific ingress and egress handling as well as providing a switch port specific timestamp, depending on the possibilities of the used hardware.
Rationale:	Switch frame management control is essential for upper layer modules, in particular for EthTSyn, which requires port specific information associated to a time synchronization or path-delay measurement frame.
Use Case:	Global time synchronization over Ethernet





Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#), [RS_BRF_01660](#))

[SRS_Eth_00126] Independent reset of host ECU and switch hardware [

Description:	The Ethernet Switch Driver module shall allow to handle resets of the host ECU and the switch hardware independently, i.e. the reset of either one shall not require a restart of the other. To ensure the correct operation of the switch hardware, it shall be possible to verify the switch configuration and to determine if a reconfiguration is required.
Rationale:	This allows the continuous operation of the host ECU and the switch hardware.
Use Case:	Continuous operation
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

[SRS_Eth_00128] The Ethernet Switch Driver shall provide statistic counter values per port [

Description:	The Ethernet Switch Driver shall provide interfaces to read out statistic counter values (such as number of received, transmitted and dropped frames, etc.) per individual port for diagnostic analysis.
Rationale:	–
Use Case:	Diagnostics
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

4.10 SWS Ethernet State Manager (EthSM)

[SRS_Eth_00043] The Ethernet State Manager shall provide network configuration and initialization. [

Description:	The Ethernet State Manager shall provide network independent configuration and initialization interface.
---------------------	--



△

Rationale:	Hardware abstraction
Use Case:	Exchanging the used network shall be transparent and only be reflected by replacing the used State Manager.
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#), [RS_BRF_01664](#))

[SRS_Eth_00158] The Ethernet state manager shall trigger requested modes for Ethernet hardware with wake-up capability even if the requested mode has already been reached. [

Description:	The Ethernet State Manager shall re-trigger requested modes for an Ethernet network with wake-up capability even if the requested mode has already been reached
Rationale:	An Ethernet switch network need to trigger wake-up and propagate the wake-up on the whole network to signal connected ECUs about a communication request
Use Case:	Wake-up of the Ethernet switched network, due to a PNC request.
Dependencies:	[SRS_Eth_00157]
Supporting Material:	[44] OPEN ALLIANCE Sleep/Wake-up Specification Version 2.0 (Rel Feb 21, 2017)

]([RS_Main_00430](#))

[SRS_Eth_00159]{DRAFT} The Ethernet state manager shall forward sleep requests indicated by the maintained Ethernet hardware that act as communication slaves to the responsible upper layer. [

Description:	The Ethernet state manager shall forward sleep requests indicated by the maintained Ethernet hardware that act as communication slaves to the responsible upper layer
Rationale:	A communication channel that acts as a communication slave without wake-up capability could only be requested remotely (passive wake-up) by its corresponding communication master. If the used Ethernet hardware detect a sleep request, the request shall be indicated directly to the ComM
Use Case:	Ethernet communication channels which use Ethernet hardware that support of OPEN ALLIANCE Sleep/Wake-up Specification Version 2.0 (Rel Feb 21, 2017) and do NOT using network management. Only single ECU which do NOT maintain an Ethernet switch could have a communication channels which act as communication slave.
Dependencies:	–
Supporting Material:	[44] OPEN ALLIANCE Sleep/Wake-up Specification Version 2.0 (Rel Feb 21, 2017)

]([RS_Main_00430](#))

4.11 SWS UDP Network Management (UdpNm)

[SRS_Eth_00037] The UDP Network Management shall provide an interface for transmission of network management information. [

Description:	The UDP Network Management shall provide an interface to send and receive network management information over UDP.
Rationale:	Hardware abstraction
Use Case:	Exchanging the used network shall be transparent and only be reflected by replacing the used Network Management.
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#), [RS_BRF_01680](#))

[SRS_Eth_00075] The UDP Network Management shall at least provide the functionality of CAN NM. [

Description:	The UDP Network Management shall match the functionality of CAN NM. (i.e Including optional services such as Partial Networking)
Rationale:	Bus independence
Use Case:	Replacement of CAN by Ethernet and IP
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#), [RS_BRF_01680](#))

4.12 SWS Service Discovery (Sd)

[SRS_Eth_00076] The APIs of the Service Discovery module shall support any protocol [

Description:	The APIs of the Sd module shall encapsulate the use of any protocol on the wire.
Rationale:	The Sd functionality may be achieved with any number of different wire protocols.
Use Case:	–
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

4.13 Non-Functional Requirements (Qualities)

4.13.1 SWS TCP/IP Protocol Stack

[SRS_Eth_00077] The TCP/IP stack shall be implemented as independent sub-modules. [

Description:	The TCP/IP stack shall be implemented as independent sub-modules.
Rationale:	The TCP/IP protocol suite is too large to be implemented in a monolithic block.
Use Case:	–
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01784](#))

4.13.2 SWS Socket Adaptor (SoAd)

[SRS_Eth_00078] The SoAd module shall be the sole PDU interface to the TCP/IP stack [

Description:	SoAd module shall be the single PDU interface to the TCP/IP stack for AUTOSAR modules.
Rationale:	Modular design Single implementations of PDU to socket transformation
Use Case:	PDU communication via Ethernet used by multiple AUTOSAR modules.
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

4.13.3 SWS Ethernet Interface (EthIf)

[SRS_Eth_00029] Ethernet Interface shall be the single interface of all Ethernet modules to the Ethernet hardware drivers. [

Description:	The Ethernet Interface shall be the single interface for all upper modules (i.e. TCP/IP module and Ethernet State Manager) to the lower layer Ethernet hardware drivers for each Ethernet controller (Ethernet Driver) and Ethernet transceiver (Ethernet Transceiver Driver).
---------------------	--



△

Rationale:	Interface and interaction
Use Case:	Multiple software modules shall transmit and receive data through multiple Ethernet connections in a uniform way.
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

[SRS_Eth_00030] The Ethernet Interface shall be independent of the actual hardware. [

Description:	The Ethernet Interface shall provide a hardware independent interface to its upper layer modules (i.e. TCP/IP module and Ethernet State Manager).
Rationale:	Portability and reusability
Use Case:	Exchanging the used Ethernet controller and transceiver shall be transparent and only be reflected by replacing the used driver.
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

4.13.4 SWS Ethernet Driver (Eth)

[SRS_Eth_00034] An Ethernet Driver shall offer a hardware independent interface. [

Description:	An Ethernet Driver shall offer a hardware independent interface for all Ethernet controllers of the same type
Rationale:	Hardware abstraction
Use Case:	Exchanging the used Ethernet controller shall be transparent and only be reflected by replacing the used driver.
Dependencies:	–
Supporting Material:	–

](RS_BRF_01776)

4.13.5 SWS Ethernet Transceiver Driver (EthTrcv)

[SRS_Eth_00038] An Ethernet Transceiver Driver shall offer a hardware independent interface. [

Description:	An Ethernet Transceiver Driver shall offer a hardware independent interface for all Ethernet transceivers of the same type
Rationale:	Hardware abstraction
Use Case:	Exchanging the used Ethernet transceiver shall be transparent and only be reflected by replacing the used driver.
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#))

4.13.6 SWS Ethernet State Manager (EthSM)

[SRS_Eth_00041] An Ethernet State Manager shall offer network independent state handling [

Description:	An Ethernet State Manager shall offer network independent state handling. - Uninitialized - No Communication - Full Communication
Rationale:	Hardware abstraction
Use Case:	Exchanging the used network shall be transparent and only be reflected by replacing the used State Manager.
Dependencies:	–
Supporting Material:	–

]([RS_BRF_01776](#), [RS_BRF_01664](#))

4.13.7 SWS UDP Network Management (UdpNm)

[SRS_Eth_00042] A UDP Network Management shall offer network independent interface. [

Description:	A UDP Network Management shall offer network independent interface
Rationale:	Hardware abstraction
Use Case:	Exchanging the used network shall be transparent and only be reflected by replacing the used Network Management.
Dependencies:	–



△

Supporting Material:	–
-----------------------------	---

]([RS_BRF_01776](#), [RS_BRF_01680](#))

5 Requirements Tracing

The following table references the features specified in [46] and links to the fulfillments of these.

Feature	Description	Satisfied by
[RS_BRF_01056]	AUTOSAR BSW modules shall provide standardized interfaces	[SRS_Eth_00053] [SRS_Eth_00058] [SRS_Eth_00103]
[RS_BRF_01104]	AUTOSAR shall support sleep and wake-up of ECUs and buses	[SRS_Eth_00106] [SRS_Eth_00107] [SRS_Eth_00108] [SRS_Eth_00149] [SRS_Eth_00150]
[RS_BRF_01317]	AUTOSAR shall support SOME/IP	[SRS_Eth_00124]
[RS_BRF_01440]	AUTOSAR services shall support system diagnostic functionality	[SRS_Eth_00024] [SRS_Eth_00025] [SRS_Eth_00026] [SRS_Eth_00027] [SRS_Eth_00028] [SRS_Eth_00047] [SRS_Eth_00080] [SRS_Eth_00081] [SRS_Eth_00082] [SRS_Eth_00083] [SRS_Eth_00084] [SRS_Eth_00104]
[RS_BRF_01656]	AUTOSAR communication shall support XCP	[SRS_Eth_00056]
[RS_BRF_01660]	AUTOSAR communication shall support distribution and synchronization of a Global Time across different networks	[SRS_Eth_00125]
[RS_BRF_01664]	AUTOSAR communication shall support a state management of buses	[SRS_Eth_00041] [SRS_Eth_00043]
[RS_BRF_01680]	AUTOSAR communication shall support mechanism to keep a bus awake, and to be kept awake by a bus	[SRS_Eth_00037] [SRS_Eth_00042] [SRS_Eth_00075]
[RS_BRF_01716]	AUTOSAR communication shall support to aggregate multiple PDUs to one PDU dynamically	[SRS_Eth_00116]

<p>[RS_BRF_01776]</p>	<p>AUTOSAR communication shall support Ethernet</p>	<p>[SRS_Eth_00001] [SRS_Eth_00002] [SRS_Eth_00004] [SRS_Eth_00005] [SRS_Eth_00006] [SRS_Eth_00008] [SRS_Eth_00009] [SRS_Eth_00011] [SRS_Eth_00012] [SRS_Eth_00013] [SRS_Eth_00029] [SRS_Eth_00030] [SRS_Eth_00031] [SRS_Eth_00032] [SRS_Eth_00033] [SRS_Eth_00034] [SRS_Eth_00035] [SRS_Eth_00036] [SRS_Eth_00037] [SRS_Eth_00038] [SRS_Eth_00039] [SRS_Eth_00040] [SRS_Eth_00041] [SRS_Eth_00042] [SRS_Eth_00043] [SRS_Eth_00048] [SRS_Eth_00049] [SRS_Eth_00050] [SRS_Eth_00051] [SRS_Eth_00069] [SRS_Eth_00070] [SRS_Eth_00071] [SRS_Eth_00072] [SRS_Eth_00073] [SRS_Eth_00074] [SRS_Eth_00075] [SRS_Eth_00076] [SRS_Eth_00078] [SRS_Eth_00087] [SRS_Eth_00088] [SRS_Eth_00105] [SRS_Eth_00106] [SRS_Eth_00107] [SRS_Eth_00108] [SRS_Eth_00114] [SRS_Eth_00116] [SRS_Eth_00117] [SRS_Eth_00118]</p>
-----------------------	---	---

		<p>[SRS_Eth_00119] [SRS_Eth_00120] [SRS_Eth_00121] [SRS_Eth_00122] [SRS_Eth_00123] [SRS_Eth_00125] [SRS_Eth_00126] [SRS_Eth_00127] [SRS_Eth_00128] [SRS_Eth_00129] [SRS_Eth_00131] [SRS_Eth_00132] [SRS_Eth_00146] [SRS_Eth_00147] [SRS_Eth_00148] [SRS_Eth_00149] [SRS_Eth_00150]</p>
<p>[RS_BRF_01784]</p>	<p>AUTOSAR communication shall support the IP protocol stack</p>	<p>[SRS_Eth_00014] [SRS_Eth_00015] [SRS_Eth_00016] [SRS_Eth_00017] [SRS_Eth_00018] [SRS_Eth_00019] [SRS_Eth_00020] [SRS_Eth_00021] [SRS_Eth_00022] [SRS_Eth_00045] [SRS_Eth_00054] [SRS_Eth_00055] [SRS_Eth_00059] [SRS_Eth_00061] [SRS_Eth_00062] [SRS_Eth_00065] [SRS_Eth_00066] [SRS_Eth_00067] [SRS_Eth_00068] [SRS_Eth_00077] [SRS_Eth_00085] [SRS_Eth_00088] [SRS_Eth_00089] [SRS_Eth_00090]</p>

		[SRS_Eth_00091] [SRS_Eth_00092] [SRS_Eth_00093] [SRS_Eth_00094] [SRS_Eth_00095] [SRS_Eth_00096] [SRS_Eth_00097] [SRS_Eth_00098] [SRS_Eth_00099] [SRS_Eth_00100] [SRS_Eth_00101] [SRS_Eth_00102] [SRS_Eth_00103] [SRS_Eth_00109] [SRS_Eth_00110] [SRS_Eth_00111] [SRS_Eth_00112] [SRS_Eth_00113] [SRS_Eth_00115]
[RS_BRF_01920]	AUTOSAR microcontroller abstraction shall provide access to communication bus controllers	[SRS_Eth_00117]
[RS_Main_00161]	Unified Abstract Application Modeling	[SRS_Eth_00134] [SRS_Eth_00135] [SRS_Eth_00136] [SRS_Eth_00137]
[RS_Main_00190]	Non-AUTOSAR Software Integration	[SRS_Eth_00133] [SRS_Eth_00138] [SRS_Eth_00139] [SRS_Eth_00140] [SRS_Eth_00141]
[RS_Main_00260]	Runtime Diagnostics Means	[SRS_Eth_00138] [SRS_Eth_00139] [SRS_Eth_00140] [SRS_Eth_00141]
[RS_Main_00280]	Standardized Automotive Communication Protocols	[SRS_Eth_00142] [SRS_Eth_00143] [SRS_Eth_00144] [SRS_Eth_00145]
[RS_Main_00420]	AUTOSAR shall use established software standards and consolidate de-facto standards for basic software functionality	[SRS_Eth_00133]
[RS_Main_00430]	AUTOSAR shall support established automotive communication standards	[SRS_Eth_00108] [SRS_Eth_00142] [SRS_Eth_00143] [SRS_Eth_00144] [SRS_Eth_00145] [SRS_Eth_00151] [SRS_Eth_00152] [SRS_Eth_00153] [SRS_Eth_00154] [SRS_Eth_00155] [SRS_Eth_00156] [SRS_Eth_00157] [SRS_Eth_00158] [SRS_Eth_00159]

[RS_Main_00510]	Secure Onboard Communication	[SRS_Eth_00138] [SRS_Eth_00139] [SRS_Eth_00140] [SRS_Eth_00141]
[RS_Main_01008]	AUTOSAR shall provide secure communication with off-board entities	[SRS_Eth_00133] [SRS_Eth_00138] [SRS_Eth_00139] [SRS_Eth_00140] [SRS_Eth_00141]

6 References

References

- [1] Standardization Template
AUTOSAR_TPS_StandardizationTemplate
- [2] Glossary
AUTOSAR_TR_Glossary
- [3] Standard for Information Technology–Portable Operating System Interface (POSIX(R)) Base Specifications, Issue 7
<http://pubs.opengroup.org/onlinepubs/9699919799/>
- [4] Road vehicles – Diagnostic communication over Internet Protocol (DoIP) – Part 2: Network and transport layer requirements and services
<http://www.iso.org>
- [5] Internet Control Message Protocol
<https://rfc-editor.org/rfc/rfc792.txt>
- [6] Transmission Control Protocol
<https://rfc-editor.org/rfc/rfc793.txt>
- [7] Requirements for Internet Hosts - Communication Layers
<https://rfc-editor.org/rfc/rfc1122.txt>
- [8] Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification
<https://rfc-editor.org/rfc/rfc4443.txt>
- [9] IEEE Standard 802.1AS-30
<http://standards.ieee.org/getieee802/download/802.1AS-2011.pdf>
- [10] Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers
<https://rfc-editor.org/rfc/rfc2474.txt>
- [11] IPv6 Flow Label Specification

- <https://rfc-editor.org/rfc/rfc6437.txt>
- [12] Internet Protocol
<https://rfc-editor.org/rfc/rfc791.txt>
 - [13] An Ethernet Address Resolution Protocol: Or Converting Network Protocol Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware
<https://rfc-editor.org/rfc/rfc826.txt>
 - [14] Dynamic Configuration of IPv4 Link-Local Addresses
<https://rfc-editor.org/rfc/rfc3927.txt>
 - [15] Internet Protocol, Version 6 (IPv6) Specification
<https://rfc-editor.org/rfc/rfc2460.txt>
 - [16] Deprecation of Type 0 Routing Headers in IPv6
<https://rfc-editor.org/rfc/rfc5095.txt>
 - [17] Path MTU Discovery for IP version 6
<https://rfc-editor.org/rfc/rfc1981.txt>
 - [18] Neighbor Discovery for IP version 6 (IPv6)
<https://rfc-editor.org/rfc/rfc4861.txt>
 - [19] IPv6 Subnet Model: The Relationship between Links and Subnet Prefixes
<https://rfc-editor.org/rfc/rfc5942.txt>
 - [20] Optimistic Duplicate Address Detection (DAD) for IPv6
<https://rfc-editor.org/rfc/rfc4429.txt>
 - [21] IP Version 6 Addressing Architecture
<https://rfc-editor.org/rfc/rfc4291.txt>
 - [22] Transmission of IPv6 Packets over Ethernet Networks
<https://rfc-editor.org/rfc/rfc2464.txt>
 - [23] Default Address Selection for Internet Protocol Version 6 (IPv6)
<https://rfc-editor.org/rfc/rfc6724.txt>
 - [24] Handling of Overlapping IPv6 Fragments
<https://rfc-editor.org/rfc/rfc5722.txt>
 - [25] IPv6 Stateless Address Autoconfiguration
<https://rfc-editor.org/rfc/rfc4862.txt>
 - [26] TCP Congestion Control
<https://rfc-editor.org/rfc/rfc5681.txt>
 - [27] The NewReno Modification to TCP's Fast Recovery Algorithm
<https://rfc-editor.org/rfc/rfc6582.txt>
 - [28] Congestion Control in IP/TCP Internetworks
<https://rfc-editor.org/rfc/rfc896.txt>

- [29] The Transport Layer Security (TLS) Protocol Version 1.2
<https://rfc-editor.org/rfc/rfc5246.txt>
- [30] Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS)
<https://rfc-editor.org/rfc/rfc4492.txt>
- [31] Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)
<https://rfc-editor.org/rfc/rfc4279.txt>
- [32] User Datagram Protocol
<https://rfc-editor.org/rfc/rfc768.txt>
- [33] Dynamic Host Configuration Protocol
<https://rfc-editor.org/rfc/rfc2131.txt>
- [34] DHCP Options and BOOTP Vendor Extensions
<https://rfc-editor.org/rfc/rfc2132.txt>
- [35] The Dynamic Host Configuration Protocol (DHCP) Client Fully Qualified Domain Name (FQDN) Option
<https://rfc-editor.org/rfc/rfc4702.txt>
- [36] Dynamic Host Configuration Protocol for IPv6 (DHCPv6)
<https://rfc-editor.org/rfc/rfc3315.txt>
- [37] The Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Client Fully Qualified Domain Name (FQDN) Option
<https://rfc-editor.org/rfc/rfc4704.txt>
- [38] DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)
<https://rfc-editor.org/rfc/rfc3646.txt>
- [39] RFC 4301, Security Architecture for the Internet Protocol
- [40] RFC 4302, IP Authentication Header
- [41] RFC 4303, IP Encapsulating Security Payload (ESP)
- [42] RFC 7296, Internet Key Exchange Protocol Version 2 (IKEv2)
- [43] Road vehicles – Unified diagnostic services (UDS) – Part 5: Unified diagnostic services on Internet Protocol implementation (UDSonIP)
<http://www.iso.org>
- [44] OPEN Sleep/Wake-up Specification for Automotive Ethernet
<http://www.opensig.org/Automotive-Ethernet-Specifications/>
- [45] OPEN TC14 - Interoperability and Compliance Tests for 10BASE-T1S PHYs
<http://www.opensig.org/Automotive-Ethernet-Specifications/>
- [46] Requirements on AUTOSAR Features
AUTOSAR_RS_Features