Document Title	Requirements on System Template
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	213
Document Classification	Auxiliary

Document Version	3.2.1
Document Status	Final
Part of Release	4.1
Revision	2

Document Change History			
Date	**Version**	**Changed by**	**Description**
29.10.2013	3.2.1	AUTOSAR Release Management	Layout update.
04.02.2013	3.2.0	AUTOSAR Administration	• Added requirements [RS_SYST_00043], [RS_SYST_00044], [RS_SYST_00045], [RS_SYST_00046], [RS_SYST_00047], [RS_SYST_00048]
05.07.2012	3.1.1	AUTOSAR Administration	• Formating updated
07.06.2011	3.1.0	AUTOSAR Administration	• Added requirement [RS_SYST_00042]

30.11.2009	3.0.0	AUTOSAR Administration	• Added requirements [RS_SYST_00027], [RS_SYST_00028], [RS_SYST_00029], [RS_SYST_00030], [RS_SYST_00031], [RS_SYST_00037], [RS_SYST_00038], [RS_SYST_00039], [RS_SYST_00041] • Refined requirement SYSCT0004 with SYSCT32, SYSCT33, SYSCT34, SYSCT35, SYSCT36 • Refined requirement SYSCT0005 with [RS_SYST_00037] and [RS_SYST_00040] • Refined requirement [RS_SYST_00007], [RS_SYST_00001], [RS_SYST_00035] • Legal disclaimer revised
23.06.2008	2.1.1	AUTOSAR Administration	Legal disclaimer revised
21.11.2007	2.1.0	AUTOSAR Administration	• Added requirements [RS_SYST_00025] and [RS_SYST_00026] • Set requirement [RS_SYST_00023] to "postponed" • Document meta information extended • Small layout adaptations made
31.01.2007	2.0.0	AUTOSAR Administration	Release as a separate Document. The Specification of System Template V1.0.0 has been split into independent documents for Release 2.1 • Legal disclaimer revised • Release Notes added • "Advice for users" revised • "Revision Information" added

09.05.2005	1.0.0	AUTOSAR Administration	Initial release as part of the Specification of System Template V1.0.0

Disclaimer

Advice for users

AUTOSAR specifications may contain exemplary items (exemplary reference models, "use cases", and/or references to exemplary technical solutions, devices, processes or software).

Any such exemplary items are contained in the specifications for illustration purposes only, and they themselves are not part of the AUTOSAR Standard. Neither their presence in such specifications, nor any later documentation of AUTOSAR conformance of products actually implementing such exemplary items, imply that intellectual property rights covering such exemplary items are licensed under the same rules as applicable to the AUTOSAR Standard.

Table of Contents

References

[1] System Template
AUTOSAR_TPS_SystemTemplate

[2] Standardization Template
AUTOSAR_TPS_StandardizationTemplate

[3] Main Requirements
AUTOSAR_RS_Main

1 Scope of this document

This document collects the requirements on the System Template (SYS-T). The main goal of the System Template is the definition of a relationship between the pure Software View on the System and a Physical System Architecture with networked ECUs.

The System Template covers the following areas:

- System topology: In the system topology the logical layout of the system is described. This means it is documented which ECU is connected to which cluster or channel.
- Communication properties: The central purpose of a communication system is the exchange of frames with certain properties.
- Mapping: The mapping covers the distribution of software components to ECUs as well as the mapping of data elements that are to be exchanged between software components onto signals and frames.

The requirements collected in this document will be satisfied by the System Template specification [1]. This document implements most of the requirements stated here.

1.1 Document Conventions

The representation of requirements in AUTOSAR documents follows the table specified in [TPS_STDT_00078], see Standardization Template, chapter Support for Traceability ([2]).

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 ([2]).

1.2 Requirements Tracing

The following table references the requirements specified in [3] and links to the fulfillments of these.

Requirement	Description	Satisfied by
[RS_Main_00010]	AUTOSAR shall provide a software platform to support the development of safety related systems.	[RS_SYST_00028]
[RS_Main_00030]	AUTOSAR shall support development processes for safety related systems	[RS_SYST_00003] [RS_SYST_00008] [RS_SYST_00009] [RS_SYST_00010] [RS_SYST_00011] [RS_SYST_00016] [RS_SYST_00017] [RS_SYST_00018] [RS_SYST_00019] [RS_SYST_00020]
[RS_Main_00060]	AUTOSAR shall provide a standardized software interface for communication between Software Components	[RS_SYST_00031]
[RS_Main_00100]	AUTOSAR shall provide standardized basic software	[RS_SYST_00002] [RS_SYST_00025]
[RS_Main_00140]	AUTOSAR shall provide network independent communication mechanisms for applications	[RS_SYST_00014]
[RS_Main_00150]	AUTOSAR shall support the reallocation of Software Components	[RS_SYST_00002] [RS_SYST_00007] [RS_SYST_00008] [RS_SYST_00009] [RS_SYST_00010] [RS_SYST_00011] [RS_SYST_00016] [RS_SYST_00017] [RS_SYST_00018] [RS_SYST_00019] [RS_SYST_00020]
[RS_Main_00190]	AUTOSAR shall support interoperability with non-AUTOSAR software on the same ECU	[RS_SYST_00001]
[RS_Main_00210]	AUTOSAR shall support interoperability with non-AUTOSAR ECUs in a network	[RS_SYST_00001] [RS_SYST_00015] [RS_SYST_00043]
[RS_Main_00230]	AUTOSAR shall support network topologies including gateways	[RS_SYST_00013] [RS_SYST_00044]
[RS_Main_00300]	AUTOSAR shall provide data exchange formats to support work-share in large inter and intra company development groups	[RS_SYST_00006]
[RS_Main_00320]	AUTOSAR shall provide formats to specify all aspects necessary to integrate a Software Component on an ECU	[RS_SYST_00007] [RS_SYST_00013]
[RS_Main_00340]	AUTOSAR shall support the observance of timing requirements	[RS_SYST_00037] [RS_SYST_00040]

Requirement	Description	Satisfied by
[RS_Main_00360]	AUTOSAR shall support management of vehicle diversity	[RS_SYST_00032] [RS_SYST_00033] [RS_SYST_00034] [RS_SYST_00035] [RS_SYST_00036] [RS_SYST_00041]
[RS_Main_00430]	AUTOSAR shall support automotive communication systems	[RS_SYST_00015] [RS_SYST_00021] [RS_SYST_00022] [RS_SYST_00023] [RS_SYST_00024] [RS_SYST_00025] [RS_SYST_00029] [RS_SYST_00030] [RS_SYST_00038] [RS_SYST_00039]

2 Requirements

2.1 Category: Main Requirements

2.1.1 Compatibility between the AUTOSAR Templates

[RS_SYST_00006] Compatibility between the AUTOSAR Templates ⌈

Type:	valid
Description:	The compatibility between the AUTOSAR Templates must be guaranteed. In this context, compatibility means that each AUTOSAR template can have references to elements of another AUTOSAR template.
Rationale:	Ensuring coherence and interoperability between AUTOSAR templates.
Dependencies:	None identified.
Use Case:	Development of an in-vehicle electronic architecture (software modelling, hardware modelling and mapping constraint modelling) using the same tool chain.
Supporting Material:	–

⌋*(RS_Main_00300)*

2.2 Category: System Template Requirements

2.2.1 Legacy systems

[RS_SYST_00001] Mixed Systems (AUTOSAR/NON-AUTOSAR) ⌈

Type:	valid
Description:	System constraints, which arise through usage of mixed systems, must be treated by System Template.
Rationale:	The transition between non-AUTOSAR systems to full-AUTOSAR systems can only be achieved gradually. Furthermore, interoperability with legacy solutions must be ensured. Thus, it must be possible to have AUTOSAR and non-AUTOSAR ECUs together on the same system ("mixed" systems).
Dependencies:	None identified.
Use Case:	Gradual AUTOSAR introduction into an existing architecture e.g. it shall be possible to handle signals not originating from AUTOSAR software components.
Supporting Material:	–

⌋*(RS_Main_00190, RS_Main_00210)*

2.2.2 Basic Software and RTE Resources

[RS_SYST_00002] Basic Software Resources and RTE Resources ⌈

Type:	valid
Description:	The System Template has to cover resource requests of the basic SW and the RTE.
Rationale:	Resources of an ECU are, by their own definition, limited (RAM, ROM, CPU time, etc.). Such limitations act as constraints during the mapping process.
Dependencies:	None identified.
Use Case:	Taking into account memory limitations when allocating AUTOSAR services and features on a small ECU.
Supporting Material:	–

⌋*(RS_Main_00150, RS_Main_00100)*

2.2.3 Iterative Development

[RS_SYST_00003] Iterative Development ⌈

Type:	valid
Description:	The System Template has to support an iterative system development.
Rationale:	During the development of an AUTOSAR system, solutions found in former steps of the system design process are themselves system constraints for the next system generation steps.
Dependencies:	None identified.
Use Case:	If new functionalities are added to a vehicle project in a late development phase, the current mapping become itself a constraint for the mapping of the new SW components associated with such new functionalities.
Supporting Material:	–

⌋*(RS_Main_00030)*

2.2.4 Mapping of Software Components to ECUs

[RS_SYST_00007] Mapping of Software Components to ECUs ⌈

Type:	valid
Description:	The System Template has to describe the mapping of software components to ECUs. An optional mapping of software components to individual processing units residing in one ECU shall also be possible.
Rationale:	–
Dependencies:	None identified.

Use Case:	For safety reasons (or simply due to the experience) some specific Software Components can run only on some specific ECUs. Such "pre-mapping" is a constraint for the real mapping process.
Supporting Material:	–

⌋(*RS_Main_00320*, *RS_Main_00150*)

2.2.5 Mapping of Software Components to ECUs: Clustering

[RS_SYST_00008] SWC Cluster ⌈

Type:	valid
Description:	The System Constraint Description has to cover the clustering of SW Components. SW Component Clustering means that two SW Components cannot be divided and must be mapped to the same ECU.
Rationale:	Due to performance requirements, to safe communication requirements or simply to experience, some communication paths must be prevented to be mapped onto an external bus. Involved SW Components shall then be mapped together onto the same ECU.
Dependencies:	None identified.
Use Case:	Safe communication, which may not be carried out over a communication bus, or very strict timing requirements.
Supporting Material:	–

⌋(*RS_Main_00030*, *RS_Main_00150*)

2.2.6 Mapping of Software Components to ECUs: Separation

[RS_SYST_00009] SWC Separation ⌈

Type:	valid
Description:	The System Constraint Description has to cover the separation of SW Components. SW Component Separation means that two SW Components cannot be on the same ECU.
Rationale:	To enhance the independence of redundant SWC.
Dependencies:	None identified.
Use Case:	Two redundant Software Components, implementing safety critical functions, will not be mapped together on the same ECU because of safety requirements (of course, redundancy does not always imply SWC separation).
Supporting Material:	–

⌋(*RS_Main_00030*, *RS_Main_00150*)

2.2.7 Exclusive Mapping of SWCs to ECUs

[RS_SYST_00010] Exclusive Mapping of SWCs ⌈

Type:	valid
Description:	The System Constraint Description has to cover the exclusion of SWCs from one or more ECUs. "Exclusion" means that the SWC cannot be mapped to the ECUs it is excluded from.
Rationale:	During the mapping process it can be useful to express that a specific SWC cannot be mapped to one or more ECUs, based on ECU properties.
Dependencies:	None identified.
Use Case:	Exclusion of safety critical functions from crash exposed areas.
Supporting Material:	–

⌋(*RS_Main_00030*, *RS_Main_00150*)

2.2.8 Dedicated Mapping of SWCs to ECUs

[RS_SYST_00011] Dedicated Mapping of SWCs ⌈

Type:	valid
Description:	The System Constraint Description has to describe dedicated mapping of SWCs to one or more ECUs. "Dedicated mapping" means that the SWC can only be mapped to the ECUs it is dedicated to.
Rationale:	During the mapping process it can be useful to express that a specific SWC can be only mapped to some ECUs, based on ECU properties.
Dependencies:	None identified.
Use Case:	SWCs requiring an ECU that can guarantee full functionality for a specified time period after power down.
Supporting Material:	–

⌋(*RS_Main_00030*, *RS_Main_00150*)

2.2.9 Topology Description

[RS_SYST_00013] Topology ⌈

Type:	valid
Description:	The System Template has to describe the topology of an EE System.
Rationale:	The available communication paths limit the possible distributions of SW Components to some ECUs.
Dependencies:	None identified.
Use Case:	Mapping of SW Components being tightly linked from a functional point of view: the topology must then be known in order to avoid too long data paths.

Supporting Material:	–

⌋(*RS_Main_00320*, *RS_Main_00230*)

2.2.10 Data Segmentation

[RS_SYST_00014] Data Segmentation ⌈

Type:	valid
Description:	The System Template must provide information, which can be used for the segmenting of (application) data to more than 1 frame.
Rationale:	Data length limitations of the underlying bus technology.
Dependencies:	None identified.
Use Case:	Transmission of diagnostic data, often longer than 8 bytes, by means of 8 byte CAN frames.
Supporting Material:	–

⌋(*RS_Main_00140*)

2.2.11 Bus bandwidth

[RS_SYST_00015] Bus bandwidth ⌈

Type:	valid
Description:	The System Template shall support bandwidth calculation as a constraint for the definition of the Communication Matrix.
Rationale:	Bandwidth is a limited resource, acting as a constraint during the definition of the Communication Matrix.
Dependencies:	None identified.
Use Case:	When defining the Communication Matrix for mixed systems (AUTOSAR and non-AUTOSAR ECUs), only one part of the Communication Matrix is freely configurable using the AUTOSAR process. That means that the available bandwidth for the AUTOSAR system generator is limited by the non-AUTOSAR part of the Communication Matrix.
Supporting Material:	–

⌋(*RS_Main_00210*, *RS_Main_00430*)

2.2.12 Dedicated physical connections

[RS_SYST_00016] Dedicated physical connections ⌈

Type:	valid
Description:	The System Constraint Description shall be able to describe that a signal has to be sent over a dedicated wire, which is only used by two SW-Components (sender and receiver).
Rationale:	This technique is commonly used in current safety concepts.
Dependencies:	None identified.
Use Case:	Communication with the airbag module.
Supporting Material:	–

⌋(*RS_Main_00150*, *RS_Main_00030*)

2.2.13 Mapping of signals to the same physical line

[RS_SYST_00017] Mapping of signals to the same physical line ⌈

Type:	valid
Description:	The System Constraint Description shall be able to describe that a group of signals has to be sent via the same physical line.
Rationale:	–
Dependencies:	None identified.
Use Case:	–
Supporting Material:	–

⌋(*RS_Main_00150*, *RS_Main_00030*)

2.2.14 Mapping of signals to different physical lines

[RS_SYST_00018] Mapping of signals to different physical lines ⌈

Type:	valid
Description:	The System Constraint Description shall be able to describe, if needed, that signals between ECUs are sent via different physical lines.
Rationale:	To support hardware and information redundancy (as a mean to support fault detection and fault handling).
Dependencies:	None identified.
Use Case:	A mean to guarantee the transmission of very safety critical data, is to force the sending of redundant copies onto different physical lines.
Supporting Material:	–

⌋(*RS_Main_00150*, *RS_Main_00030*)

2.2.15 Mapping of signals to a specific physical line

[RS_SYST_00019] Mapping of signals to a specific physical line ⌈

Type:	valid
Description:	The System Constraint Description shall be able to describe that signals have to be mapped to a specific physical line.
Rationale:	Some signals have to be mapped to specific physical lines due to e.g. special performance and/or safety needs.
Dependencies:	None identified.
Use Case:	Powertrain signals have to be mapped to a high-speed bus, due to their timing requirements.
Supporting Material:	–

⌋*(RS_Main_00150, RS_Main_00030)*

2.2.16 Exclusion of signals from a specific physical line

[RS_SYST_00020] Exclusion of signals from a specific physical line ⌈

Type:	valid
Description:	The System Constraint Description shall be able to describe that signals have not to be mapped to a specific physical line.
Rationale:	Some physical lines can result unsuitable (too slow, unsafe communication protocol, etc.) for the transmission of some specific signals.
Dependencies:	None identified.
Use Case:	Most of power train signals cannot be mapped to a low speed CAN bus, due to their timing requirements.
Supporting Material:	–

⌋*(RS_Main_00150, RS_Main_00030)*

2.2.17 ECU Communication via CAN

[RS_SYST_00021] ECU Communication via CAN ⌈

Type:	valid
Description:	The System Template has to cover the system communication via CAN Bus.
Rationale:	CAN is widely used in the automotive systems.
Dependencies:	None identified.
Use Case:	Development of a complete, multi-networked, in-vehicle electronic architecture.
Supporting Material:	–

⌋(*RS_Main_00430*)

2.2.18 ECU Communication via LIN

[RS_SYST_00022] ECU Communication via LIN ⌈

Type:	valid
Description:	The System Template has to cover the system communication via LIN.
Rationale:	LIN is widely used in the automotive systems.
Dependencies:	None identified.
Use Case:	Development of a complete, multi-networked, in-vehicle electronic architecture.
Supporting Material:	–

⌋(*RS_Main_00430*)

2.2.19 ECU Communication via MOST

[RS_SYST_00023] ECU Communication via MOST ⌈

Type:	valid
Description:	The System Template has to cover the system communication via MOST.
Rationale:	MOST is going to become a standard communication protocol in the automotive industry.
Dependencies:	None identified.
Use Case:	Development of a complete, multi-networked, in-vehicle electronic architecture.
Supporting Material:	–

⌋(*RS_Main_00430*)

2.2.20 ECU Communication via FlexRay

[RS_SYST_00024] ECU Communication via FlexRay ⌈

Type:	valid
Description:	The System Template has to cover the system communication via FlexRay.
Rationale:	FlexRay is going to become a standard communication protocol in the automotive industry.
Dependencies:	None identified.
Use Case:	Development of a complete, multi-networked, in-vehicle electronic architecture.
Supporting Material:	–

⌋(*RS_Main_00430*)

2.2.21 Derivation of COM Stack Configuration Parameters from the System Template

[RS_SYST_00025] Derivation of COM Stack Configuration Parameters from the System Template ⌈

Type:	valid
Description:	The System Template shall enable the configuration of the Com Stack of the ECU. It handles those parameters that are necessary to describe the inter-ECU communication. Configuration parameters local to an ECU are not in the scope of the System Template.
Rationale:	All ECUs connected in one communication cluster needs to be configured consistently.
Dependencies:	None identified.
Use Case:	Generate Base ECU Configuration from ECU Extract
Supporting Material:	–

⌋(*RS_Main_00430*, *RS_Main_00100*)

2.2.22 ASAM FIBEX compatibility

[RS_SYST_00026] Fibex compatibility ⌈

Type:	valid
Description:	Whenever there is a considerable overlap between the System Template and the ASAM FIBEX Standard, the System Template shall adopt the structures of the ASAM FIBEX Standard.
Rationale:	The System Template will benefit from FIBEX as an established proven standard.
Dependencies:	None identified.
Use Case:	Facilitate the adoption of the System Template into existing tools which deal with the FIBEX Standard.
Supporting Material:	ASAM FIBEX

⌋

2.2.23 ECU Extract generation rules

[RS_SYST_00027] ECU Extract generation rules ⌈

Type:	valid

Description:	The ECU Extract is derived from a System Description. The specification for generating the ECU Extract shall be detailed enough to enable semantically unambiguous generation of this artifact.
Rationale:	Tool interoperability requires unambiguous description of the ECU Extract.
Dependencies:	None identified.
Use Case:	Generate Base ECU Configuration from ECU Extract
Supporting Material:	–

⌋

2.2.24 IPdu End-to-End Communication Protection support

[RS_SYST_00028] IPdu End-to-End Communication Protection support ⌈

Type:	valid
Description:	The System Template shall enable to select E2E protection settings for IPdus.
Rationale:	Protect communication between COM modules.
Dependencies:	None identified.
Use Case:	Transmission of safety-related data on a single channel without redundancy.
Supporting Material:	–

⌋*(RS_Main_00010)*

2.2.25 Dynamic length signals

[RS_SYST_00029] Dynamic length signals ⌈

Type:	valid
Description:	The System Template shall support a definition of dynamic length signals. A Signal shall have either a static length or its length should vary up to some statically defined maximum. Signals with a maximum length are called dynamic length signals.
Rationale:	Dynamic length signals can change size during run time.
Dependencies:	None identified.
Use Case:	–
Supporting Material:	OSEK COM

⌋*(RS_Main_00430)*

2.2.26 Dynamic length IPdus

[RS_SYST_00030] Dynamic length IPdus ⌈

Type:	valid
Description:	The System Template shall support a definition of IPdus that contain dynamic length signals.
Rationale:	Dynamic length IPdus can change size during run time.
Dependencies:	[RS_SYST_00029]
Use Case:	The Network Layer and the Data Link Layer are capable of transmitting and receiving both fixed and dynamic-length I-Pdus as determined by the Interaction Layer.
Supporting Material:	OSEK COM

⌋*(RS_Main_00430)*

2.2.27 Distribution of Application and Vehicle Mode Requests

[RS_SYST_00031] Distribution of Application and Vehicle Mode Requests ⌈

Type:	valid
Description:	The System Template shall support the distribution of application and vehicle mode requests to all affected ECUs.
Rationale:	A Mode Requester is an entity that requests modes from a Mode Manager by sending some data via a port with a Mode Request interface. The Mode Manager receives the incoming information, arbitrates the requests and decides upon a resulting mode.
Dependencies:	None identified.
Use Case:	Depending on Vehicle and Application Modes, the BSW modes may change, e.g. the communication needs of an Application may cause a change in the BSW Mode of a communication network.
Supporting Material:	–

⌋*(RS_Main_00060)*

2.2.28 Topology variants

[RS_SYST_00032] Topology variants ⌈

Type:	valid
Description:	The System Template shall provide the means to describe topology variants with optional/alternative ECUs and communication clusters.
Rationale:	In a product line approach different product variants can be realized by a common core topology with only a few varying topology nodes.
Dependencies:	None identified.

Use Case:	In a product line two different product variants HIGH and LOW use the same common core topology with the difference, that the product variant HIGH requires an additional ECU.
Supporting Material:	–

⌋(*RS_Main_00360*)

2.2.29 Software-to-ECU mapping variants

[RS_SYST_00033] Software-to-ECU mapping variants ⌈

Type:	valid
Description:	The System Template shall provide the means to describe alternative mappings of software components to ECUs.
Rationale:	In order to reach different specific characteristics of the overall system for products within a product line a different mapping of software components is used.
Dependencies:	[RS_SYST_00007],[RS_SYST_00008],[RS_SYST_00009],[RS_SYST_00010], [RS_SYST_00011],[RS_SYST_00013]
Use Case:	In a product line two different product variants HIGH and LOW use the same common software architecture but define a different mapping to the network topology.
Supporting Material:	–

⌋(*RS_Main_00360*)

2.2.30 Timing Variants

[RS_SYST_00034] Timing Variants ⌈

Type:	valid
Description:	The System Template shall provide the means to describe alternative timing properties (e.g. trigger type, period, priority) and timing constraints (e.g. latency, age).
Rationale:	Due to a different software-to-ECU mapping the timing properties and constraints for the transmission of signals can vary.
Dependencies:	None identified.
Use Case:	A PDU is transmitted cyclically for two different product variants HIGH and LOW with a period of 10ms for the variant HIGH and 20ms for the variant LOW.
Supporting Material:	–

⌋(*RS_Main_00360*)

2.2.31 Data mapping variants

[RS_SYST_00035] Data mapping variants ⌈

Type:	valid
Description:	The System Template shall provide the means to describe data mapping Variants.
Rationale:	Variants in the Software Component Description have an impact on the data mapping which is described in the System Template.
Dependencies:	None identified.
Use Case:	A DataElement exists only in one product variant HIGH.
Supporting Material:	–

⌋(*RS_Main_00360*)

2.2.32 Communication variants

[RS_SYST_00036] Communication variants ⌈

Type:	valid
Description:	The System Template shall provide the means to describe communication variants, such as alternative signal-to-PDU mappings, alternative communication paths, and alternative signal and PDU properties (e.g. data type, data length).
Rationale:	To optimize the communication matrix for different product variants which use the same network topology the description of communication variants in the System Template is an essential prerequisite.
Dependencies:	[RS_SYST_00032],[RS_SYST_00035]
Use Case:	A signal is transmitted for two different product variants HIGH and LOW with the byte order LittleEndian for the variant HIGH and the byte order BidEndian for the variant LOW.
Supporting Material:	–

⌋(*RS_Main_00360*)

2.2.33 Timing properties

[RS_SYST_00037] Timing properties ⌈

Type:	valid
Description:	The System Template shall provide the means to describe the timing properties of a system's dynamics, which are determined by the consumption of computation, communication, and other hardware resources.

Rationale:	The description of timing properties in the System Template is an essential prerequisite for the analysis and validation of a system's timing behavior or its prediction early in the process.
Dependencies:	None identified.
Use Case:	Analysis and validation of timing behavior, early prediction of modification impacts, support for hardware dimensioning, system configuration optimization
Supporting Material:	–

⌋(*RS_Main_00340*)

2.2.34 Support of SAE J1939 Protocol Features

[RS_SYST_00038] Support of SAE J1939 Protocol Features ⌈

Type:	valid
Description:	The System Template has to cover the system communication via SAE J1939.
Rationale:	SAE J1939 protocol is an industry standard that is used in automotive systems.
Dependencies:	None identified.
Use Case:	Development of a complete, multi-networked, in-vehicle electronic architecture.
Supporting Material:	–

⌋(*RS_Main_00430*)

2.2.35 ECU Communication via Ethernet

[RS_SYST_00039] ECU Communication via Ethernet ⌈

Type:	valid
Description:	The System Template has to cover the system communication via Ethernet.
Rationale:	Ethernet is going to become a standard communication protocol in the automotive industry.
Dependencies:	None identified.
Use Case:	Development of a complete, multi-networked, in-vehicle electronic architecture.
Supporting Material:	–

⌋(*RS_Main_00430*)

2.2.36 Timing constraints

[RS_SYST_00040] Timing constraints ⌈

Type:	valid
Description:	The System Template shall provide the means to describe the timing constraints of a system's dynamics, which are determined by the consumption of computation, communication, and other hardware resources.
Rationale:	The description of timing constraints in the System Template is an essential prerequisite for the analysis and validation of a system's timing behavior or its prediction early in the process.
Dependencies:	None identified.
Use Case:	Analysis and validation of timing behavior, early prediction of modification impacts, support for hardware dimensioning, system configuration optimization
Supporting Material:	–

⌋(*RS_Main_00340*)

2.2.37 Variants in ECU Extract

[RS_SYST_00041] Variants in ECU Extract ⌈

Type:	valid
Description:	The ECU Extract shall support variability of elements taken over or derived during the transformation from the System Description.
Rationale:	Data mapping and communication variants (see [RS_SYST_00035], [RS_SYST_00036]) may have to be preserved in artifacts, which are generated out of a System Description (e.g. ECU-Extract), if the binding time is at a later point in the process.
Dependencies:	None identified.
Use Case:	Pdu Layout is postbuild configurable during the ECU Configuration, variability needs to be visible at build time.
Supporting Material:	–

⌋(*RS_Main_00360*)

2.2.38 Support for Partial Networking

[RS_SYST_00042] Support for Partial Networking ⌈

Type:	valid
Description:	System Template shall support the definition of partial network clusters, the mapping of virtual function clusters to partial network clusters and the wakeup information of individual ECUs.
Rationale:	System Template shall contain all system relevant parameters required for the configuration of a partial network.
Dependencies:	None identified.
Use Case:	Describe the size and location of the system wide partial network cluster vector.

Supporting Material:	–

⌋

2.2.39 Communication via Complex Drivers

[RS_SYST_00043] Communication via Complex Drivers ⌈

Type:	valid
Description:	System Template shall support the Pdu-based communication via Complex Drivers.
Rationale:	It shall be possible to describe the Complex Driver Pdus that are transmitted over the network.
Dependencies:	None identified.
Use Case:	A new BSW module is used above the PduR, e.g a Diagnostic Service.
Supporting Material:	–

⌋*(RS_Main_00210)*

2.2.40 Description of custom bus systems

[RS_SYST_00044] Description of custom bus systems ⌈

Type:	valid
Description:	System Template shall support the integration of custom bus systems on the topology level.
Rationale:	It shall be possible to describe the complete network topology of a vehicle with a System Description.
Dependencies:	None identified.
Use Case:	Alternative communication technologies (e.g. I2C, USB, serial line) are integrated as Complex Drivers
Supporting Material:	–

⌋*(RS_Main_00230)*

2.2.41 Co-existing System artifacts in the same model

[RS_SYST_00045] Co-existing System artifacts in the same model ⌈

Type:	valid

Description:	The System Template shall provide means to describe different System Extracts, which can co exist with the complete system description and each other in the same model.
Rationale:	–
Dependencies:	None identified.
Use Case:	The OEM hands a system extract over to the supplier, as a formal "requirement" specification. The supplier extends and refactors this system extract. In the next development cycle the OEM hands over an updated system extract to the supplier. The supplier updates his system extract according to the update of the OEM.
Supporting Material:	[RS_METH_00077]

⌋

2.2.42 Different views on the system's SWC-structure

[RS_SYST_00046] Different views on the system's SWC-structure ⌈

Type:	valid
Description:	The System Template shall provide means to describe the different views on the SWC-structure and the mapping between their elements.
Rationale:	–
Dependencies:	None identified.
Use Case:	Different views on the SWC-structure at an OEM: Functional view (independent of ECUs) and ECU-topological view
Supporting Material:	[RS_METH_00078], [RS_METH_00079]

⌋

2.2.43 Network and physical representation on signal level

[RS_SYST_00047] Network and physical representation on signal level ⌈

Type:	valid
Description:	The System Template shall provide means to describe the physical representation and the network representation of signals.
Rationale:	–
Dependencies:	None identified.
Use Case:	–
Supporting Material:	–

⌋

2.2.44 CAN with Flexible Data-Rate

[RS_SYST_00048] CAN with Flexible Data-Rate ⌈

Type:	valid
Description:	The System Template shall support the CAN FD protocol.
Rationale:	CAN FD increases the bandwidth of a CAN network and allows payload larger than 8 byte.
Dependencies:	None identified.
Use Case:	Development of a complete, multi-networked, in-vehicle electronic architecture
Supporting Material:	–

⌋

3 Change History

3.1 Change History for AUTOSAR 4.0.1 against 3.1.5

3.1.1 Removed SRS Items

Number	Heading
[RS_SYST_00004]	Variant Handling
[RS_SYST_00005]	Timing Requirements

Table 3.1: Removed Specification Items in 4.0.1

3.1.2 Changed SRS Items

Number	Heading
[RS_SYST_00001]	Mixed Systems (AUTOSAR/NON-AUTOSAR)
[RS_SYST_00007]	Mapping of Software Components to ECUs

Table 3.2: Changed Specification Items in 4.0.1

3.1.3 Added SRS Items

Number	Heading
[RS_SYST_00027]	ECU Extract generation rules
[RS_SYST_00028]	IPdu End-to-End Communication Protection support
[RS_SYST_00029]	Dynamic length signals
[RS_SYST_00030]	Dynamic length IPdus
[RS_SYST_00031]	Distribution of Application and Vehicle Mode Requests
[RS_SYST_00032]	Topology variants
[RS_SYST_00033]	Software-to-ECU mapping variants
[RS_SYST_00034]	Timing variants
[RS_SYST_00035]	Data mapping variants
[RS_SYST_00036]	Communication variants
[RS_SYST_00037]	Timing properties
[RS_SYST_00038]	Support of SAE J1939 Protocol Features
[RS_SYST_00039]	ECU Communication via Ethernet
[RS_SYST_00040]	Timing constraints
[RS_SYST_00041]	Variants in ECU Extract

Table 3.3: Added Specification Items in 4.0.1

3.2 Change History for AUTOSAR 4.0.2 against 4.0.1

3.2.1 Removed SRS Items

N/A

3.2.2 Changed SRS Items

N/A

3.2.3 Added SRS Items

N/A

3.3 Change History for AUTOSAR 4.0.3 against 4.0.2

3.3.1 Removed SRS Items

N/A

3.3.2 Changed SRS Items

N/A

3.3.3 Added SRS Items

Number	Heading
[RS_SYST_00042]	Support for Partial Networking

Table 3.4: Added Specification Items in 4.0.3

3.4 Change History for AUTOSAR 4.1.1 against 4.0.3

3.4.1 Removed SRS Items

N/A

3.4.2 Changed SRS Items

N/A

3.4.3 Added SRS Items

3.4.4 Added SRS Items

Number	Heading
[RS_SYST_00043]	Communication via Complex Device Drivers
[RS_SYST_00044]	Description of custom bus systems
[RS_SYST_00045]	Co-existing System artifacts in the same model
[RS_SYST_00046]	Different views on the system's SWC-structure
[RS_SYST_00047]	Network and physical representation on signal level
[RS_SYST_00048]	CAN with Flexible Data-Rate

Table 3.5: Added Specification Items in 4.1.1