Document Title	Requirements on Watchdog Driver
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	197
Document Classification	Auxiliary

Document Version	2.0.5
Document Status	Final
Part of Release	3.2
Revision	1

Document Change History

Date	Version	Changed by	Change Description
23.03.2011	2.0.5	AUTOSAR Administration	Legal disclaimer revised
23.06.2008	2.0.4	AUTOSAR Administration	Legal disclaimer revised
31.10.2007	2.0.3	AUTOSAR Administration	• Document meta information extended • Small layout adaptations made
24.01.2007	2.0.2	AUTOSAR Administration	• “Advice for users” revised • “Revision Information” added
28.11.2006	2.0.1	AUTOSAR Administration	Legal disclaimer revised
14.03.2006	2.0.0	AUTOSAR Administration	Release as a separate document. The SRS SPAL V1.0.0 has been split into 15 independent documents for Release 2.0
11.07.2005	1.0.0	AUTOSAR Administration	Initial release as a part of the SRS SPAL V1.0.0

Disclaimer

Advice for users

AUTOSAR Specification Documents 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 Specification Documents for illustration purposes only, and they themselves are not part of the AUTOSAR Standard. Neither their presence in such Specification Documents, 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

1 Scope of this document

This document specifies requirements on the module Watchdog Driver.

Constraints

First scope for specification of requirements on basic software modules are systems which are not safety relevant. For this reason safety requirements are assigned to medium priority.

2 How to read this document

Each requirement has its unique identifier starting with the prefix “BSW” (for “Basic Software”). For any review annotations, remarks or questions, please refer to this unique ID rather than chapter or page numbers!

2.1 Conventions used

In requirements, the following specific semantics are used (taken from Request for Comment RFC 2119 from the Internet Engineering Task Force IETF)

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 described in RFC 2119. Note that the requirement level of the document in which they are used modifies the force of these words.

- MUST: This word, or the terms "REQUIRED" or "SHALL", mean that the definition is an absolute requirement of the specification.
- MUST NOT: This phrase, or the phrase „SHALL NOT“, means that the definition is an absolute prohibition of the specification.
- SHOULD: This word, or the adjective "RECOMMENDED", mean 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" mean 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 marketplace 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, MUST 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, MUST be prepared to interoperate with another implementation, which does not include the option (except, of course, for the feature the option provides.)

2.2 Requirement structure

Each module specific chapter contains a short functional description of the Basic Software Module. Requirements of the same kind within each chapter are grouped under the following headlines (where applicable):

Functional Requirements:
- Configuration (which elements of the module need to be configurable)
- Initialisation
- Normal Operation
- Shutdown Operation
- Fault Operation
- ...

Non-Functional Requirements:
- Timing Requirements
- Resource Usage
- Usability
- Output for other WPs (e.g. Description Templates, Tooling,...)
- ...

3 Acronyms and abbreviations

Acronyms and abbreviations that have a local scope are not contained in the AUTOSAR glossary. These must appear in a local glossary.

Acronym / Abbreviation	***Description:***
CS	Chip select
DIO	Digital Input Output
ECU	Electric Control Unit
EOL	End Of Line Often used in the term 'EOL Programming' or 'EOL Configuration'
HIS	Herstellerinitiative Software
ICU	Interrupt Capture Unit
MAL	Old name of Microconroller Abstraction Layer (replaced by MCAL because 'MAL' is a french term meaning 'bad')
MCAL	Microconroller Abstraction Layer
MCU	Microcontroller Unit
MMU	Memory Management Unit
Master	A device controlling other devices (slaves, see below)
Slave	A device beeing completely controlled by a master device
NMI	Non maskable interrupt
OS	Operating System
PLL	Phase Locked Loop
PWM	Pulse Width Modulation
RX	Reception (in the context of bus communication)
SPAL	The name of this working group
SFR	Special Function Register
RTE	Runtime environment
WP	Work Package
STD	Standard
REQ	Requirement
UNINIT	Uninitialized (= not initialized)

As this is a document from professionals for professionals, all other terms are expected to be known.

4 Requirement Specification

4.1 Internal Watchdog Driver

4.1.1 Functional Overview

The Internal Watchdog Driver controls the internal watchdog timer of the MCU. It offers the trigger functionality and a mode select service.

4.1.2 Functional Requirements

4.1.2.1 Configuration

4.1.2.1.1 [BSW12015] Configuration of watchdog modes

Initiator:	BMW
Date:	03.05.2004
Short Description:	Configuration of watchdog modes
Type:	New
Importance:	High
Description:	The watchdog driver shall allow the static configuration of watchdog modes. A watchdog mode shall minimally consist of the desired watchdog period. Any MCU specific parameter can be added. Further explanation: Each watchdog mode has the same set of parameters, values will differ.
Rationale:	For mode switching.
Use Case:	Another mode parameters could be: • selection of window / timeout mode • timeout reaction (reset or NMI)
Dependencies:	[BSW12018] Watchdog mode selection service
Conflicts:	--
Supporting Material:	BMW Specification MCAL V1.0a, REQ MAL31.1.2

4.1.2.2 Initialization

4.1.2.2.1 [BSW12105] Watchdog initialization

Initiator:	WP4.2.2.1.12
Date:	09.06.2004
Short Description:	Watchdog initialization
Type:	New
Importance:	High
Description:	The watchdog driver shall provide an initialization service that allows the selection of one of the statically configured watchdog modes.
Rationale:	Basic functionality
Use Case:	--
Dependencies:	--
Conflicts:	--
Supporting Material:	--

4.1.2.2.2 [BSW12106] Prohibit disabling of watchdog

Initiator:	WP4.2.2.1.12
Date:	09.06.2004
Short Description:	Prohibit disabling of watchdog
Type:	New
Importance:	High
Description:	The watchdog initialization service and the watchdog mode selection service must not allow the disabling of the watchdog. This requirement is only applicable for safety relevant systems. For that reason, this feature shall be statically configurable (by a preprocessor switch)
Rationale:	Avoid the presence of code sequences in a safety relevant ECU that disable the watchdog.
Use Case:	Usage within safety relevant systems.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

4.1.2.3 Normal Operation

4.1.2.3.1 [BSW12018] Watchdog mode selection service

Initiator:	BMW
Date:	03.05.2004
Short Description:	Watchdog mode selection service
Type:	New
Importance:	High
Description:	The watchdog driver shall provide a service for selecting the watchdog mode: • Fast mode (mandatory) • Slow mode (optional) • Off (optional)
Rationale:	Allow adaptation of watchdog behavior to ECU state.
Use Case:	Allow switching of different timeout periods for start-up and run mode: • ECU Start-up mode: Slow mode (long timeout period) • ECU Run mode: Fast mode (short timeout period)
Dependencies:	[BSW12015] Configuration of watchdog modes
Conflicts:	--
Supporting Material:	It is not required for each microcontroller to provide all modes. Some watchdogs do not allow mode changes once they have been set up.

4.1.2.3.2 [BSW12019] Watchdog trigger service

Initiator:	BMW
Date:	09.06.2004
Short Description:	Watchdog trigger service
Type:	Changed
Importance:	High
Description:	The watchdog driver shall provide a watchdog trigger service. This service shall allow data exchange with the watchdog device (to and from)
Rationale:	Basic functionality

Use Case:	The data exchange can be used with complex watchdogs that provide a password mechanism (e.g. for use in safety relevant systems).
Dependencies:	--
Conflicts:	--
Supporting Material:	BMW Specification MCAL V1.0a, REQ MAL31.X

4.1.2.4 Shutdown Operation

A Deinit function is not provided for the watchdog driver due to safety reasons and because most watchdogs do not allow a deactivation. Thus, [BSW12163] Driver module deinitialization is not valid for this module.

4.2 External Watchdog Driver

4.2.1 Functional Overview

The External Watchdog Driver controls an external hardware watchdog. It offers the trigger functionality and a mode select service. It has the same functional scope like the internal watchdog driver.

4.2.2 Functional Requirements

4.2.2.1 General

4.2.2.1.1 [BSW12165] Functional scope

Initiator:	BMW
Date:	07.07.2004
Short Description:	Functional scope
Type:	New
Importance:	High
Description:	For an external watchdog driver the same requirements shall apply like for an internal watchdog driver.
Rationale:	Make no functional differences between internal and external watchdog. Keep the functional scope the same.
Use Case:	--
Dependencies:	Requirements on internal watchdog driver
Conflicts:	--
Supporting Material:	--

4.2.2.2 Configuration

4.2.2.2.1 [BSW12166] SPI channel configuration

Initiator:	WP4.2.2.1.12
Date:	07.07.2004
Short Description:	SPI channel configuration
Type:	Changed (generalized)
Importance:	High
Description:	A driver for an external SPI watchdog shall allow the static configuration of the required SPI parameters. Those parameters are specified by the SPI Handler specification.
Rationale:	Basic configuration of SPI access
Use Case:	Use the SPI watchdog driver together with other SPI device drivers on the same SPI bus.
Dependencies:	--
Conflicts:	--
Supporting Material:	AUTOSAR SWS SPI Handler

4.2.3 Non-Functional Requirements (Qualities)

4.2.3.1 [BSW12167] Common Watchdog API

Initiator:	BMW
Date:	13.05.2004
Short Description:	Common Watchdog API
Type:	New
Importance:	High
Description:	The external watchdog driver shall have a semantically identical API as an internal watchdog driver.
Rationale:	Ease control of watchdogs by the Watchdog Manager. Keep handling of internal and external Watchdogs similar.
Use Case:	Use the same Watchdog manager with an internal or with an external watchdog driver.
Dependencies:	Requirements on internal watchdog driver
Conflicts:	--
Supporting Material:	--

4.2.3.2 [BSW12168] Microcontroller independency

Initiator:	BMW
Date:	13.05.2004
Short Description:	Microcontroller independency
Type:	New
Importance:	High
Description:	The source code of the external watchdog driver shall be independent from the underlying microcontroller.
Rationale:	Reuse of external watchdog driver across multiple microcontrollers
Use Case:	Example: The same external watchdog driver for an SPI watchdog device can be used on a NEC V850 and on a Renesas M16C without any modification using the standardized SPI Handler interface.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

5 References

5.1 Deliverables of AUTOSAR

[1] List of Basic Software Modules
AUTOSAR_BasicSoftwareModules.pdf

[2] Layered Software Architecture
AUTOSAR_LayeredSoftwareArchitecture.pdf

[3] General Requirements on Basic Software Modules
AUTOSAR_SRS_General.pdf

[4] General Requirements on SPAL
AUTOSAR_SRS_SPAL_General.pdf

5.2 Related standards and norms

[5] HIS API I/O Driver Specification
www.automotive-his.de/results/
API_IODriver_2.1.3.pdf