Safe Watchdog Interface

User Manual

Version: 1.9.4
Date: 21.05.2014
Document number: D-MSP-M-70-006

TTTech Automotive GmbH

Schoenbrunner Str. 7, A-1040 Vienna, Austria, Tel. + 43 1 585 34 34-0, Fax +43 1 585 34 34-90, support@tttech-automotive.com

Table of Contents

1 Introduction

The Safe Watchdog Interface (S-WdgIf) is a part of the Safe Watchdog Manager Stack.

Note: This is a user manual and does *not* cover safety-related topics. For safety-related projects that need to fulfill ISO 26262 requirements, refer to the *Safe Watchdog Interface Safety Manual* [5] [20].

Note: The < > placeholder used in this document stands for the *infix* part of the names of the Watchdog driver functions to which the S-WdgIf interfaces. Depending on the version of the used AUTOSAR environment, the S-WdgIf can consist of the following:

- In AUTOSAR 4.0 compatible environment, the S-WdgIf consists of the vendor ID and device name strings, where vendo ID is the ID of the vendor of the Watchdog driver and device name is the name of the configured Watchdog driver device .
- In AUTOSAR 3.1 compatible environment, the S-WdgIf consists of the device name string, where device name is the name of the configured Watchdog driver device
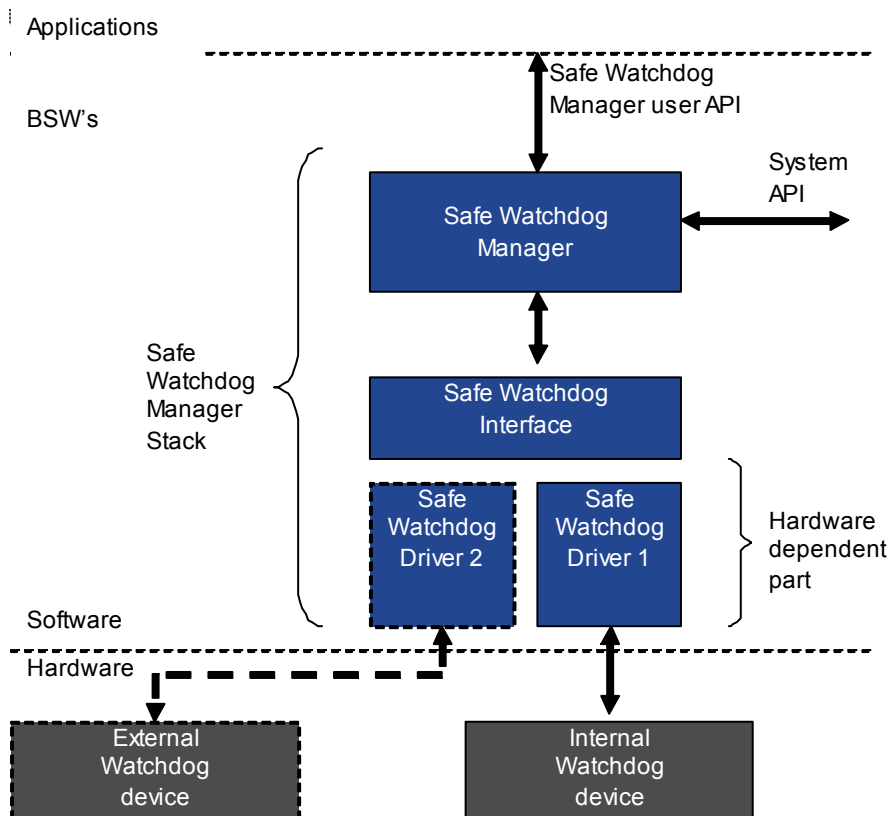
2 Safe Watchdog Interface

2.1 Basic Functionality of the S-WdgIf

The S-WdgIf was developed according AUTOSAR version 4.0.1 [2] [20] and adapted for the AUTOSAR 3.1.4 environment [1] [20]. The S-WdgIf is compatible with both AUTOSAR versions, but not fully compliant. For the deviations, see section *Deviations from the AUTOSAR Watchdog Interface* [5].

The S-WdgIf is designed to be integrated into an AUTOSAR 3.1.4 and 4.0.1 system. However, it is not restricted to this AUTOSAR version. The software module can also be integrated into other versions of AUTOSAR and other system software architectures if the integration related requirements listed in the Safe Watchdog Interface Safety Manual [5] [20] are met.

The S-WdgIf provides a standard interface to all the configured watchdog devices. The Safe Watchdog Manager (S-WdgM) accesses each configured watchdog through its Device Index.

The S-WdgIf is hardware-independent and abstracts one or more Safe Watchdog Driver modules for the S-WdgM. The S-WdgM calls the S-WdgIf with a device index parameter (D I). It is translated by the S-WdgIf into a S-Wdg driver instance. If necessary, additional driver parameters are provided.

Figure 1 shows the layered structure of the S-WdgM. The attached watchdog devices can be internal, external, or both.

Applications

Safe Watchdog
Manager user API

BSW's

System
API

Safe
Watchdog
Manager
Stack

Safe Watchdog
Manager

Safe Watchdog
Interface

Safe
Watchdog
Driver 2

Safe
Watchdog
Driver 1

Hardware
dependent
part

Software

Hardware

External
Watchdog
device

Internal
Watchdog
device

. 1:

2.2   Extensions to the AUTOSAR Watchdog Interface

The S-WdgIf implements and extends the AUTOSAR Watchdog Interface module. It implements the following two interfaces to the Watchdog Driver:

- AUTOSAR-compatible interface
```
<     > S          C          ()
<     > S  M    ()
```

- TTTech-compatible interface
```
<     > S                ()
<     > S  M    ()
```

Switching between the two interfaces with the watchdog driver is done by setting the parameter ____I____A____D____A____ [6] in a correspoding way. For details, see section Integration with Fully AUTOSAR Compliant Drivers [5] and the description of parameter ____I____A____D____A____ [6] in section Configuration Parameters for the S-WdgIf [5].

Function   I G   C   () is an additional extension to AUTOSAR, providing access to the corresponding S-Wdg function <   > G   C   (), if supported by the hardware and if the configuration parameter ____I__I_____C____R____ [7] is set in a correspoding way. For details, see the description of parameter ____I__I_____C____R____ [7] in section Configuration Parameters for the

S-WdgIf[5].

2.3 Deviations from the AUTOSAR Watchdog Interface

For safety reasons, the S-WdgIf module should not depend on external modules. This is why the AUTOSAR module Development Error Tracer (DET) is called in the presence of development errors only if the preprocessor switch `DGIF_DE_ERROR_DE_EC`[8] is set to `S_D_ON`.

The S-WdgIf calls function `A_D_R_E_()`[15] in order to report detected DET errors instead of calling function `D_R_E_()`[15] specified in AUTOSAR. For details, see section Expected Interfaces[15].

2.4 Integration with Fully AUTOSAR Compliant Drivers

In order to integrate the S-WdgIf with a fully AUTOSAR-compliant watchdog driver set the configuration parameter `I_A_D_A`[6] to `S_D_ON`. This will result in the following:

- The AUTOSAR `<_>S_M_()` is called out of `I_S_M_()`.
- The `<_>S_C_()` is called out of `I_S_C_()`.
- The `<_>S_C_()` is called out of `I_S_()` as well, however, the parameter WindowStart is not passed.

Note: If the S-WdgM is the caller of the S-WdgIf (i.e., function `I_S_()` is used to service the watchdog device), then the parameter `S_(_M_S_)` has no effect, because it cannot be passed to an AUTOSAR-compliant driver. It is then good practice to set it to 0, because this would be the functional meaning of its absence.

For more information about the configuration parameter `I_A_D_A`[6], see section Configuration Parameters for the S-WdgIf[5].

2.5 Configuration Parameters for the S-WdgIf

Parameter Name	`I_D_I`
Path	`I_/_I_D_/_I_D_I`
Group	Watchdog device
Type	Integer
Range	`0...65535`
Compatibility	AUTOSAR
Description	Represents the Watchdog Device ID so that it can be referenced by the S-WdgM.

Parameter Name	`I D      R`
Path	`I /    I D    /    I D      R`
Group	Watchdog device
Type	Reference
Range	n.a.
Compatibility	AUTOSAR
Description	Reference to the Watchdog driver of this Watchdog Device.

Parameter Name	`I D  E      D`
Parameter Name (Embedded Code)	`DGIF DE   ERROR DE EC`
Path	`I /    I G      /    I D  E      D`
Group	Preprocessor
Type	`B`
Range	`/`
Compatibility	AUTOSAR
Description	Pre-processor switch for enabling the development error reporting.

Parameter Name	`I        I    A`
Parameter Name (Embedded Code)	`DGIF  ERSION INFO API`
Path	`I /    I G      /    I        I    A`
Group	Preprocessor
Type	`B`
Range	`/`
Compatibility	AUTOSAR
Description	Pre-processor switch to enable/disable the service returning the version information.

Parameter Name	I A D A
Parameter Name (Embedded Code)	DGIF SE A OSAR DR API
Path	I / I G / I A D A
Group	Preprocessor
Type	Boolean
Range	/
Compatibility	TTTech
Description	Pre-processor switch to enable/disable the use of fully AUTOSAR-compliant driver API functions.

Parameter Name	I I C R
Parameter Name (Embedded Code)	DGIF IN ERNAL ICK CO N ER
Path	I / I G / I I C
Group	Preprocessor
Type	Reference
Range	n.a.
Compatibility	TTTech
Description	If this parameter references a driver which implements an internal tick counter then the function I I C () is compiled and can be used by the S-WdgM. Otherwise the API is not compiled. Note: If a driver is selected, then it must support the internal tick counter, and its parameter I C must be set to R E.

2.6 Configuring the S-WdgIf in the ECU Description File

Apart from the preprocessor options the user must add a   I D   container to the S-WdgIf module in the ECU description file. This container has two attributes: I D   I   shall be set to zero, and __I D   R__ [6] points must be selected so that it points to the Watchdog driver being used.

2.7 Preprocessor Options

Option	Description
`DGIF DE  ERROR DE EC`	Enables or disables calls to DET in case of development errors. Corresponds to ECU description option `I  D  E    D` [6]. • `S D ON`: Development errors are checked and reported to DET • `S D OFF`: Development errors are checked but not reported to DET
`DGIF  ERSION INFO API`	Enables the version info API for compiling (can be disabled in order to save resources). Corresponds to ECU description option `I    I    A` [8]. • `S D ON`: The S-WdgIf API for version information is provided. • `S D OFF`: The S-WdgIf API for version information is not provided.
`DGIF IN ERNAL  ICK CO N ER`	Enables or disables the usage of an internal tick counter in the S-WdgIf. Corresponds to ECU description option `I I        C    R` [8]. If a driver is referenced the usage of an internal tick counter is enabled, otherwise disabled.
`DGIF  SE A  OSAR DR  API`	Enables or disables the use of fully AUTOSAR-compliant driver API functions. • `S D ON`: AUTOSAR-compliant driver API functions are used. Note: The parameter `S` (`M        S`) of the S-WdgM configuration is then ignored. Therefore, it is good practice to set it to 0. • `S D OFF`: TTTech driver API functions are used.

2.8 File Structure

Figure 2 gives an overview of the file structure of the S-WdgIf.
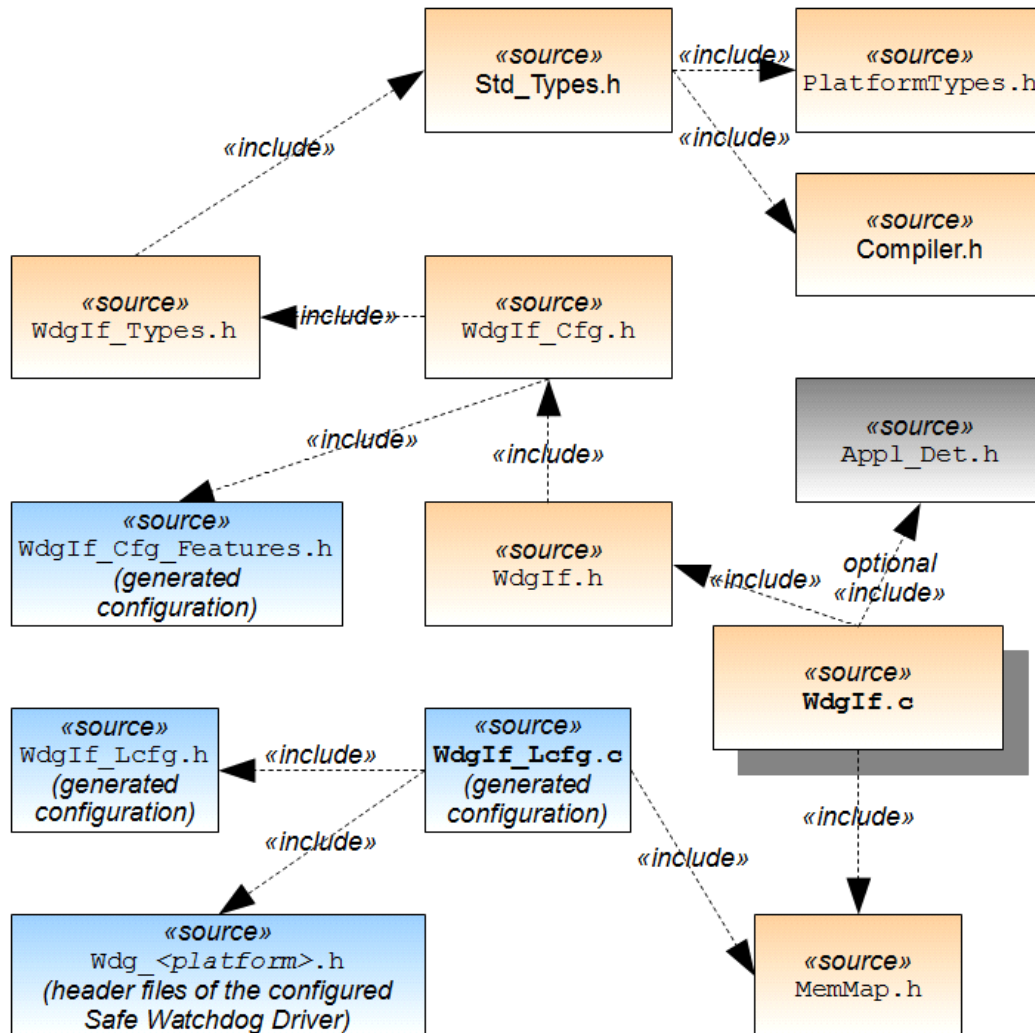
. 2:   -

The implementation of the S-WdgIf module is in the file   I . . File   I .
provides an interface to the upper layer, the S-WdgM, and includes the header file
      I . . The header files   I   . and S   . are included
indirectly through the   I . .

File	Description
I .	S-WdgIf and common Safe Watchdog Stack type definitions
I C .	Pre-compile time definitions
M M .	Is included directly in the module implementation files to organize code, data and constants in the memory.
A D .	A D . is included instead of D . , because the reporting of development errors is not done by directly calling the DET service D R E () [15], but by calling the user-defined service A D R E () [15]. Note: This service could just be a direct call to the external module DET, but could also perform more complex operations such as switching the OS context before calling DET.
I L . and I L .	These files contain the generated configuration.
I C F .	Is generated and contains all preprocessor options for the S-WdgIf module.

3   API Description

This section describes the types, functions and interfaces that are imported or provided by the S-WdgIf software layer.

3.1   Type Definitions

This section describes the types of the parameters passed to the API functions of the S-WdgIf.

Name	I I F	
Type	S	

Elements	S R (*
	    S M) (8,
	  I M)
		Pointer to the platform-specific s M function

	(* S) (8, 16, 16)	Pointer to the platform-specific S function
Description	Provides pointers to the platform-specific APIs	

Name	I I F P D	
Type	S	
Elements	I I F * F	Pointers to the platform-specific functions
	8 I	Index of the physical watchdog instance within this platform
Description	Connects platform-dependent functions to a physical watchdog in order to allow several watchdogs of the same platform to work simultaneously (e.g., external watchdogs).	

Name	I I	
Type	S	
Elements	8 N O	Number of watchdogs supported in the S-WdgIf
	I I F P D * F P D	Reference to the functions and physical watchdog indices
	# (DGIF IN ERNAL ICK CO N ER	Function pointer to the G C driver function if the

	== S D ON) 32 (* G C) ( ) #	internal tick counter is switched on
Description	Main S-Wdglf configuration structure.	

Name	`I   M`
Type	`E`
Range	▪ `DGIF OFF MODE`: Watchdog disabled ▪ `DGIF SLO  MODE`: Long timeout period (slow triggering) ▪ `DGIF FAS  MODE`: Short timeout period (fast triggering)
Description	Mode of the Watchdog

3.2 S-Wdglf Functions

Syntax	`S   R           I  S  M` `(   8 D    I   ,   I  M      M   )`
Service ID [hex]	`0 01`
Sync/Async	Synchronous
Reentrant?	No

Parameters (in)	▪ `D    I    `: Identifies the watchdog instance.
	▪ `M     `: One of the following statically configured modes:
	  `DGIF OFF MODE`: Watchdog disabled
	  `DGIF SLO  MODE`: Long timeout period (slow triggering)
	  `DGIF FAS  MODE`: Short timeout period (fast triggering)

Parameters (in/out)	none
Parameters (out)	none
Return value	▪ `E  OK`: API finished successfully. ▪ `E NO  OK`: An error occurred during execution.
Description	This function maps the `S  M` service to the corresponding physical watchdog implementation according to the parameter `D   I   `.

Syntax	S R I S C (8 D I , 16)
Service ID[hex]	0 02
Sync/Async	Synchronous
Reentrant?	No
Parameters (in)	▪ D I : Identifies the watchdog instance. ▪ : Timeout value in milliseconds for setting the trigger counter.
Parameters (in/out)	none
Parameters (out)	none
Return value	▪ E OK: API finished successfully. ▪ E NO OK: An error occurred during execution.
Description	This function maps the S C service to the corresponding physical watchdog implementation according to the parameter D I .

Syntax	S R I S (8 D I , 16 S , 16)
Service ID[hex]	0 04
Sync/Async	Synchronous
Reentrant?	No
Parameters (in)	▪ D I : Identifies the watchdog instance. ▪ S : Minimum time until next watchdog service is allowed in milliseconds. ▪ : Timeout value in milliseconds for setting the trigger counter.
Parameters (in/out)	none
Parameters (out)	none
Return value	▪ E OK: API finished successfully. ▪ E NO OK: An error occurred during execution.

Description	This function maps the `S` service to the corresponding physical watchdog implementation according to the parameter `D` `I` .

Syntax	`I` `G` `I` `(S` `I` `*` `I` `P` `)`
Service ID[hex]	`0` `03`
Sync/Async	Synchronous
Reentrant?	No
Parameters (in)	`I` `P` : Pointer to where to store the version information of this module.
Parameters (in/out)	none
Parameters (out)	none
Return value	
Description	This function is implemented as a macro and returns the version information about this module. This function is only enabled if the preprocessor switch `DGIF` `ERSION INFO API`[8] is `S` `D` `ON`.

Syntax	`32` `I` `G` `C` `(` `)`
Service ID[hex]	none
Sync/Async	Synchronous
Reentrant?	No
Parameters (in)	`C` : Pointer to where to store the tick counter value provided by the driver.
Parameters (in/out)	none
Parameters (out)	none
Return value	The current hardware tick counter of type `32`.
Description	This function returns the current hardware tick counter.

3.3 Expected Interfaces

This section describes the expected interfaces to external modules used by the S-WdgIf and their activation conditions.

Function	Description
`A    D    R    E    ()`	If the preprocessor option `DGIF_DE  ERROR_DE_EC` [8] is set to `S_D_ON`, the S-WdgIf references the function `A    D    R    E    ()` of the DET API. Note: If the pre-compiler option `DGIF_DE  ERROR_DE_EC` is set to `S_D_OFF`, the S-WdgIf is self-contained and does not call any functions from external modules.

4   S-WdgIf Configuration

This section describes the configuration of the S-WdgIf.

4.1   Link Time Configuration

The link time configuration for the S-WdgIf is configured in the ECU description file, e.g., by a tool such as DaVinci Configurator Pro . Basically, link time configuration contains all information needed for mapping each underlying watchdog driver to a device index. The configuration can be generated by the S-WdgIf configuration generator, described in section The S-WdgIf Configuration Generator[>16].

4.2   S-WdgIf Configuration Generator

The S-WdgIf generator is a Microsoft Windows console application that can be launched from a command prompt window by entering the command
        `I   C   G   .`   . The S-WdgIf generator reads the S-WdgIf module information from the AUTOSAR ECU description file (`*.`   ) and generates configuration structures for the S-WdgIf.
Note: Alternatively, you can use the DaVinci Configurator from Vector Informatik GmbH.

To use the S-Wdg generator, enter the following command in a command prompt window:

```
    I  C  G  .                    <EC -DESC-FILE> <O  P  -DIR>
<BS MD DIR>
```

	Description
`--`	Show the application version number and exits.
`- /--`	Shows this help message and exits.

Parameter	Description
`<EC -DESC-FILE>`	The ECU description file (`*.`). It is generated by a development tool (e.g.,from the DaVinci tool chain).
`<O  P  -DIR>`	The destination folder for the generated output. You must specify this parameter.
`<BS MD DIR>`	The directory in which the Configuration Generator recursively searches for the BSWMD file(s) that describe the used Watchdog drivers.

The respective configuration for the S-WdgIf is exported to two files:
- `I  L  .`
- `I  L  .`

4.3 Error Messages

The generator will show an error message in the command prompt window and quit if something goes wrong during configuration generation.

4.3.1 Basic Errors

Error No.	Error Message
1	B .
2	C EC % .
3	C % /% .
4	C % .
5	F .
6	M % % .
7	M M .

4.3.2 Semantic Errors

Error No.	Error Message
1007	M – R ICLK .
1008	N – % .
1009	% : B DGIF SLO MODE : I = % > % = S M M .
1010	% : B DGIF SLO MODE : I S = % > % = S M M .
1011	% : B DGIF FAS MODE : I = % > % = F M M .
1012	% : B DGIF FAS MODE : I S = % > % = F M M .
1013	% : B DGIF FAS MODE : I S = % >= % = I .
1014	DGIF OFF MODE % .
1022	/ S C / R K – .
1023	/ S C / F K – .
1024	M . (A EC ' A OSAR ML).
1030	/ / S C / SIRC – .
1032	C % .

		DEFINI ION-REF	.../ .
1033	IF	‰ .	
		DEFINI ION-REF	
	... I / I D / I D R .		
1041	N	‰ .	
	.. I / I D .		
1042	D ‰	.	
	.../ I D I .		
1046	E	'	
1047	E	M M . .	
1067	N D M	.	
1074		M	
	(‰ H) R I (‰ H)		
	: 0 < <= / 2.		
1077	MPC5643L MC33904 I		
	= 12 .		
1078	MPC5643L MC33904 S		
	= 6 .		
1079	O M M		
	‰ M C		
	I (‰) ‰ .		
1080	O M M		
	‰ M S		
	S (‰) ‰ .		
1087	1		
	(‰)		
1088	‰		
	I		
	I I C R .		
1089	(‰) I		
	I I C R		
	.		
1117	/ / S C /S C K - .		
1118	MPC56 :		
	S M M C .		

5 Abbreviations

Abbreviation	Description
API	Application Programming Interface
AUTOSAR	AUTOSAR (AUTomotive Open System ARchitecture) is a worldwide development partnership of car manufacturers, suppliers and other companies from the electronics, semiconductor and software industry.
DEM	Diagnostic Event Manager
DET	Development Error Tracer
ECU	Electronic Control Unit
MCU	Microcontroller Unit
S-Wdg	Safe Watchdog Driver (implementation by TTTech)
S-WdgIf	Safe Watchdog Interface (implementation by TTTech)
S-WdgM	Safe Watchdog Manager (implementation by TTTech)

6 References

[1]   AUTOSAR, *Specification of Watchdog Interface*. V. 2.2.2, Rel. 3.1, Rev. 1

[2]   AUTOSAR, *Specification of Watchdog Interface*. V. 2.3.0, Rel. 4.0, Rev. 1

[3]   TTTech Automotive GmbH, *Safe Watchdog Manager*, Safety Manual, V. 2.0.2

[4]   TTTech Automotive GmbH, *Safe Watchdog Manager*, User Manual, V. 1.8.0

[5]   TTTech Automotive GmbH, *Safe Watchdog Interface*, Safety Manual, V. 1.8.0

Index

- A -

- B -

- C -

- D -

- E -

- F -

- I -

- L -

- M -

- P -

- S -

- T -

- W -