

Software Safety Requirements and Architecture

Lane Assistance

**Document Version: 1.0**

**Template Version 1.0, Released on 2017-06-21**



# Document history

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Editor | Description |
| 24.06.2018 | 1.0 | Malgorzata Plonka | Initial version |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 

# Table of Contents

[Document history](#_1t3h5sf)

[Table of Contents](#_pyp8fd3vpmy3)

[Purpose](#_c1lz2bx22jid)

[Inputs to the Software Requirements and Architecture Document](#_hjpgfzcjxim1)

[Technical safety requirements](#_lc6owg9q3amb)

[Refined Architecture Diagram from the Technical Safety Concept](#_db3hhee81tpq)

[Software Requirements](#_pul4igmpfvr0)

[Refined Architecture Diagram](#_b7fyegncumoz)

# Purpose

The purpose of this document is to define requirements for the software components at component level to identify problems on software design and architecture that could cause a violation of safety goals.

Software safety requirements are derived from technical safety requirements. They are more specific than technical safety requirements.

The software requirements are analyzed and implemented by the software engineers.

# Inputs to the Software Requirements and Architecture Document

## Technical safety requirements

Technical Safety Requirements related to Functional Safety Requirement 01-01 are:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Technical Safety Requirement** | **ASIL** | **Fault Tolerant Time Interval** | **Architecture Allocation** | **Safe State** |
| Technical  Safety  Requirement  01 | The LDW safety component  shall ensure that the amplitude  of ‘LDW\_Torque\_Request’ sent  to the ‘Final electronic power  steering Torque’ component is  below *Max\_Torque\_Amplitude*. | C | 50 ms | LDW Safety | LDW torque set to zero. |
| Technical  Safety  Requirement  02 | When the LDW is deactivated, the ‘LDW Safety’ software module shall send a signal to the  Car Display ECU to turn on a warning signal. | C | 50 ms | LDW Safety | LDW torque set to zero. |
| Technical  Safety  Requirement  03 | When the failure is detected  by the LDW function, it shall  deactivate the LDW feature and  the ‘LDW\_Torque\_Request’ shall be set to zero. | C | 50 ms | LDW Safety | LDW torque set to zero. |
| Technical  Safety  Requirement  04 | The validity and integrity of the data transmission for  ‘LDW\_Torque\_Request’ signal shall be ensured. | C | 50 ms | Data Transmission Integrity Check | N/A |
| Technical  Safety  Requirement  05 | Memory test shall be conducted at startup of the EPS ECU to check for any faults in memory. | A | Ignition  cycle | Memory Test | LDW torque set to zero. |

## Refined Architecture Diagram from the Technical Safety Concept

# 

# Software Requirements

**Lane Departure Warning (LDW) Amplitude Malfunction Software Requirements:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Technical Safety Requirement** | **ASIL** | **Fault Tolerant Time Interval** | **Allocation to Architecture** | **Safe State** |
| Technical  Safety  Requirement  01 | The LDW safety component shall ensure that the amplitude of the LDW\_Torque\_Request sent to the Final Electronic Power Steering Torque component is below Max\_Torque\_Amplitude | C | 50 ms | LDW safety | LDW torque set to zero. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Software Safety Requirement | ASIL | Allocation Software Elements | Safe State |
| Software  Safety  Requirement  01-01 | The input signal “Primary\_LDW\_Torq\_Req” shall  be read and pre-processed to  determine the torque request  coming from the “Basic/Main  LAFunctionality” SW Component.  Signal “processed\_LDW\_Torq\_Req” shall be generated at the end  of the processing. | C | LDW\_SAFETY\_INPUT\_P  ROCESSING | N/A |
| Software Safety Requirement 01-02 | In case the  “processed\_LDW\_Torq\_Req”  signal has a value greater than “Max\_Torque\_Ampltide\_LDW”  (maximum allowed safe  torque), the torque signal  “limited\_LDW\_Torq\_Req” shall  be set to 0, else “limited\_LDW\_Torq\_Req”  shall take the value of  “processed\_LDW\_Torq\_Req”. | C | TORQUE\_LIMITER | “limited\_LDW\_T  orq\_Req” = 0  (Nm=Newtonmeter) |
| Software Safety Requirement 01-03 | The “limited\_LDW\_Torq\_Req”  shall be transformed into a  signal “LDW\_Torq\_Req” which is  suitable to be transmitted  outside of the LDW Safety  component (“LDW Safety”) to  the “Final EPS Torque” component. | C | LDW\_SAFETY\_OUTPUT  \_GENERATOR | LDW\_Torq\_Req  = 0 (Nm) |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Technical Safety Requirement** | **ASIL** | **Fault Tolerant Time Interval** | **Allocation to Architecture** | **Safe State** |
| Technical  Safety  Requirement  02 | The validity and integrity of the data transmission for LDW\_Torque\_Request signal shall be ensured | C | 50 ms | Data  Transmission  Integrity Check | N/A |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Software Safety Requirement** | **ASIL** | **Allocation Software Elements** | **Safe State** |
| Software Safety Requirement 02-01 | Any data to be transmitted  outside the LDQ Safety  component (‘LDW Safety’)  including ‘LDW\_Torque\_Req’ and ‘activation\_status’ shall be  protected by an E2E protection mechanism. | C | E2C Calc | LDW\_Torq\_Req  = 0 (Nm) |
| Software Safety Requirement 02-02 | The E2E protection protocol  shall contain and attach the  control data: alive counter (SQC)  and CRC to the data to be transmitted. | C | E2ECalc | LDW\_Torq\_Req  = 0 (Nm) |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Technical Safety Requirement** | **ASIL** | **Fault Tolerant Time Interval** | **Allocation to Architecture** | **Safe State** |
| Technical  Safety  Requirement  03 | As soon as a failure is detected by the LDW function, it shall deactivate the LDW feature and the LDW\_Torque\_Request shall be set to zero | C | 50 ms | LDW safety | LDW\_Torqu  e\_Output=  0 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Software Safety Requirement** | **ASIL** | **Allocation Software Elements** | **Safe State** |
| Software Safety Requirement03-01 | Each of the SW elements shall  output a signal to indicate any  error which is detected by the  element. Error signal =  error\_status\_input  (LDW\_SAFETY\_INPUT\_PROC  ESSING),  error\_status\_torque\_limiter  (TORQUE\_LIMITER),  error\_status\_output\_gen  (LDW\_SAFETY\_OUTPUT\_GEN  ERATOR) | C | All | N/A |
| Software Safety Requirement03-02 | A software element shall  evaluate the error status of all  the other software elements and  in case any one of them indicates an error, it shall deactivate theLDW feature (“activation\_status”=0) | C | LDW\_SAFETY  \_ACTIVATION | Activation\_status = 0  (LDW function  deactivated) |
| Software Safety Requirement03-03 | In case of no errors from the  software elements, the status of  the LDW feature shall be set to  activated (“activation\_status”=1) | C | LDW\_SAFETY  \_ACTIVATION | N/A |
| Software Safety Requirement03-04 | In case an error is detected by  any of the software elements, it  shall set the value of its  corresponding torque to 0 so  that “LDW\_Torq\_Req” is set to 0 | C | All | LDW\_Torq\_Req = 0 |
| Software Safety Requirement03-05 | Once the LDW functionality has  been deactivated, it shall stay  deactivated till the time the  ignition is switched from off to on again. | C | LDW\_SAFETY  \_ACTIVATION | Activation\_status = 0  (LDW function  deactivated) |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Technical Safety Requirement** | **ASIL** | **Fault Tolerant Time Interval** | **Allocation to Architecture** | **Safe State** |
| Technical  Safety  Requirement  04 | As soon as the LDW function deactivates the LDW feature, the LDW Safety software block shall send a signal to the car display ECU to turn on a warning light | C | 50 ms | LDW Safety | LDW\_Torqu  e\_Output=  0 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Software Safety Requirement** | **ASIL** | **Allocation Software Elements** | **Safe State** |
| Software Safety Requirement 04-01 | When the LDW function is  deactivated (activation\_status  set to 0), the activation\_status  shall be sent to the car  display ECU. | C | LDW\_SAFETY\_ACTIVATION, CarDisplay ECU | N/A |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Technical Safety Requirement** | **ASIL** | **Fault Tolerant Time Interval** | **Allocation to Architecture** | **Safe State** |
| Technical  Safety  Requirement  05 | Memory test shall be conducted at startup of the EPS ECU to check for any faults in memory | A | Ignition  cycle | Memory test | LDW\_Torqu  e\_Output=  0 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Software Safety Requirement** | **ASIL** | **Allocation Software Elements** | **Safe State** |
| Software Safety Requirement 05-01 | A CRC verification check over  the software code in the Flash  memory shall be done every  time the ignition is switched from  off to on to check for any  corruption of content. | A | MEMORYTEST | Activation\_status = 0 |
| Software Safety Requirement 05-02 | Standard RAM tests to check  the data bus, address bus and  device integrity shall be done  every time the ignition is  switched from off to on (E.g.walking 1s test, RAM  pattern test. Refer RAM and  processor vendor recommendations) | A | MEMORYTEST | Activation\_status = 0 |
| Software Safety Requirement 05-03 | The test result of the RAM or  Flash memory shall be indicated  to the LDW\_Safety component  via the “test\_status” signal | A | MEMORYTEST | Activation\_status = 0 |
| Software Safety Requirement 05-04 | In case any fault is indicated via  the “test\_status” signal the  INPUT\_LDW\_PROCESSING  shall set an error on  error\_status\_input (=1) so that  the LDW functionality is  deactivated and the LDW Torque is set to 0 | A | LDW\_SAFETY  \_INPUT\_PROC  ESSING | Activation\_status = 0 |

# 

# Refined Architecture Diagram

