



# Software Safety Requirements and Architecture

## Lane Assistance

Document Version: [Version]

Template Version 1.0, Released on 2017-06-21



# Document history

| Date       | Version | Editor     | Description      |
|------------|---------|------------|------------------|
| 27.02.2020 | 1.0     | Uwe Ehmann | Initial Document |
|            |         |            |                  |
|            |         |            |                  |
|            |         |            |                  |
|            |         |            |                  |

## Table of Contents

[Document history](#)

[Table of Contents](#)

[Purpose](#)

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

[Technical safety requirements](#)

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

[Software Requirements](#)

[Refined Architecture Diagram](#)

## Purpose

The purpose is to develop software requirements from the technical safety requirements and develop measures to ensure functional safety.

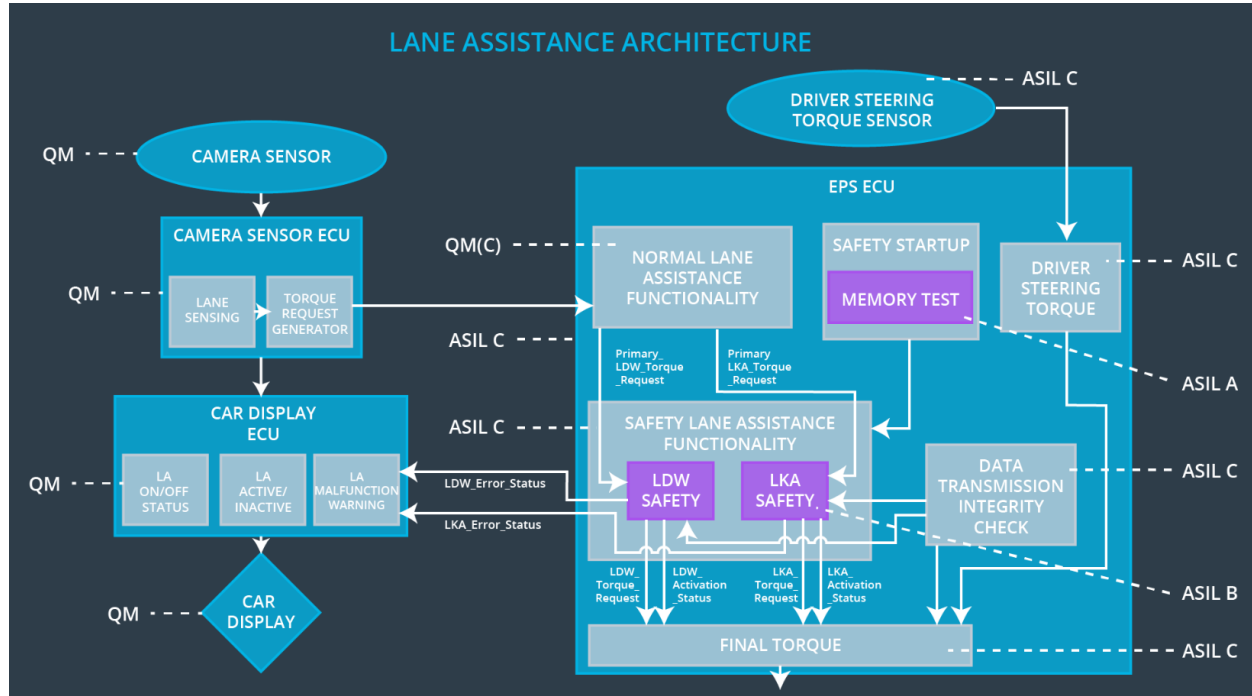
# Inputs to the Software Requirements and Architecture Document

## 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 the LDW_TORQUE_REQUEST sent to the 'Final electronic power steering Torque' component is below MAX_TORQUE_AMPLITUDE. | C    | 50 ms                        | EPS ECU – LDW Safety    | LDW Torque Request Frequency shall be set to zero |
| Technical Safety Requirement 02 | 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                        | EPS ECU – LDW Safety    | LDW Torque Request Frequency shall be set to zero |
| 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                        | EPS ECU – LDW Safety    | LDW Torque Request Frequency shall be 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                        | EPS ECU – LDW Safety    | LDW Torque Request Frequency shall be set to zero |
| Technical Safety Requirement 05 | Memory test shall be conducted at startup of the EPS ECU to check for any faults in memory.  | C    | Ignition cycle               | EPS ECU – LDW Safety    | LDW Torque Request Frequency shall be 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                        | EPS ECU – LDW Safety       | LDW Torque Request Amplitude shall be set to zero. |

| ID                                | Software Safety Requirement  | A<br>S<br>I<br>L | Allocation Software Elements    | Safe State                                     |
|-----------------------------------|--|------------------|---------------------------------|--|
| Software Safety Requirement 01-01 | The input signal "Primary_LDW_Torq_Req" shall be read and preprocessed 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<br>ROCESSING | N/A  |
| Software Safety Requirement 01-02 | In case the "processed_LDW_Torq_Req" signal has a value greater than "Max_Torque_Amplitude_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_Torq_Req" = 0<br>(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. Also see SofSafReq02-01 and SofSafReq02-02                      | C                | LDW_SAFETY_OUTPUT<br>GENERATOR  | LDW_Torq_Req = 0 (Nm)                          |

| ID                                       | Technical Safety Requirement   | A<br>S<br>I<br>L | Fault<br>Tolerant<br>Time<br>Interval | Allocation to<br>Architecture           | Safe State |
|--|--|------------------|---------------------------------------|---|------------|
| Technical<br>Safety<br>Requirement<br>02 | The validity and integrity of the data transmission for LDW_Torque_Request signal shall be ensured | C                | 50 ms                                 | Data<br>Transmission<br>Integrity Check | n/a        |

| ID   | Software Safety Requirement  | A<br>S<br>I<br>L | Allocation Software<br>Elements | Safe State               |
|--|--|------------------|---------------------------------|--------------------------|
| Software<br>Safety<br>Requirement<br>02-01 | Any data to be transmitted outside of the LDW Safety component ("LDW Safety") including "LDW_Torque_Req" and "activation_status" (see SofSafReq03-02) shall be protected by an End2End(E2E) protection mechanism | C                | E2ECalc                         | LDW_Torq_R<br>eq= 0 (Nm) |
| Software<br>Safety<br>Requirement<br>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_R<br>eq= 0 (Nm) |

| ID                                       | Technical Safety Requirement  | A<br>S<br>I<br>L | Fault<br>Tolerant<br>Time<br>Interval | Allocation to<br>Architecture | Safe State                             |
|--|---|------------------|---------------------------------------|-------------------------------|--|
| Technical<br>Safety<br>Requirement<br>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                                 | EPS ECU –<br>LDW Safety       | LDW torque<br>output is set<br>to zero |

| ID   | Software Safety Requirement  | A<br>S<br>I<br>L | Allocation<br>Software<br>Elements | Safe State   |
|--|--|------------------|------------------------------------|--|
| Software<br>Safety<br>Requirement<br>03-01 | Each of the SW elements shall output a signal to indicate any error which is detected by the element. Error signal =<br>error_status_input(LDW_SAFETY_INPUT_PROCESSING),<br>error_status_torque_limiter(TORQUE_LIMITER),<br>error_status_output_gen(LDW_SAFETY_OUTPUT_GENERATOR) | C                | All                                | n/a  |
| Software<br>Safety<br>Requirement<br>03-02 | A software element shall evaluate the error status of all the other software elements and in case any 1 of them indicates an error, it shall deactivate the LDW feature (“activation_status”=0)  | C                | LDW_SAFETY_ACTIVATIO               | Activation_status = 0<br>(LDW function<br>deactivated) |
| Software<br>Safety<br>Requirement<br>03-03 | 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_ACTIVATIO               | n/a  |
| Software<br>Safety<br>Requirement<br>03-04 | 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<br>Safety<br>Requirement<br>03-05 | 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_<br>ACTIVATIO | Activation_status = 0<br>(LDW function<br>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 torque output is set to zero |

| 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  | A<br>S<br>I<br>L | Fault Tolerant Time Interval | Allocation to Architecture | Safe State                       |
|---------------------------------|---|------------------|------------------------------|----------------------------|----------------------------------|
| Technical Safety Requirement 05 | Memory test shall be conducted at start up of the EPS ECU to check for any faults in memory | A                | Ignition cycle               | Memory test                | LDW torque output is set to zero |

| ID                                | Software Safety Requirement  | A<br>S<br>I<br>L | 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                | MEMORYTES<br>T               | 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                | MEMORYTES<br>T               | 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                | MEMORYTES<br>T               | 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 LDWTorque is set to 0                        | A                | MEMORYTES<br>T               | Activation_status = 0 |

# Refined Architecture Diagram

