L2 Functional Safety

Z. Gu 2021

Acknowledgement: some materials based on Udacity Self Driving Car Engineer Nanodegree MOOC
https://www.udacity.com/course/self-driving-car-engineer-nanodegree--nd013

Outline

- Introduction to functional safety
- Hazard Analysis and Risk Assessment (HARA)
- Functional Safety Concept
- Technical Safety Concept
- Functional Safety for Hardware and Software

Vocabulary: Fault, Failure, Hazard and Risk.

- A fault is when something inappropriate happens to the system, such as a defect or unexpected behavior.
- A fault can lead to a failure, which is synonymous with a malfunction. Failure means that the system has stopped working properly. The system is no longer doing what it is supposed to do.
- A failure could lead to a hazard. A hazard is a situation that could cause injury to a person or harm a person's health. If a system fails, the situation is potentially hazardous.
- Hazards have different levels of risk.
- Examples:
  - If a resistor in your car radio hardware breaks (fault), the radio won't turn on (failure). You won't get to listen to music, but that won't cause you bodily injury or harm. So this is a hazard with low risk.
  - If a resistor in the power steering hardware breaks (fault), the power steering could fail (failure). If you were driving at high speed, then you might get injured quite badly. So this is a hazard with high risk.

Notions of Safety in the Automotive Context

- Functional Safety
  - HW/SW not working according to specification
- Safety of intended functionality
  - Hazardous situation due to lack of specification
- Safety in use
  - Hazardous misuse/misunderstanding
- Active/passive safety
  - Accidents and their impact
- Security
  - Inherently affects safety

Functional Safety

- The term "functional" comes from a branch of systems engineering called requirements engineering. Systems engineering separates requirements into:
  - Functional requirements - what your system is supposed to do; in other words, the system's functions, with the form X system shall do Y.
    - e.g., "The turn signal system shall turn on an indicator light telling the driver that the system is active".
  - Non-functional requirements - how the system should behave, e.g., how reliable is the system? With the form X system shall be Y.
    - e.g., "The turn signal system shall be available when the vehicle ignition switch is in the on position“.
    - It does not specify the function of the turn signal system, only its availability, hence it is not a functional requirement
- Functional safety looks at what happens when the system does something that it was not supposed to do, which is called a malfunction. You will be adding new engineering requirements to the vehicle design in order to ensure safe systems.

Automotive Safety vs. Automotive Functional Safety

- Automotive functional safety is only concerned with E/E (Electrical/Electronic) system malfunctions. It is just one part of overall vehicle safety.
  - Vehicles have a number of different systems including E/E, hydraulic, mechanical and chemical. There are a variety of other automotive standards covering topics about safety that do not concern E/E system.

Functional Safety vs. Nominal Performance

- Functional safety does not look at nominal performance. Nominal performance issues can lead to safety problems, but nominal performance is not part of the functional safety standard.
  - e.g., nominal performance of an automatic braking system could be “a vehicle traveling 60 miles per hour should come to a complete stop in 5 seconds.”
  - Functional safety does not determine whether or not the vehicle brakes within 5 seconds. Instead, functional safety would look at malfunctions like
    - If the braking function is not engaged when it is supposed to
    - If the braking function is engaged when it isn't supposed to
    - if the system brakes too hard causing injury to the driver.

Quiz: Which of the following is part of a functional safety analysis?

- 1. A mechanical seatbelt restraint system could fail, and a passenger would be injured in an accident
- 2. Testing to see if a backup camera actually meets its technical specifications of a depth perception of 20 feet
- 3. There is a potential that a software error could cause the parking brake to engage at high speeds
- 4. a car battery might catch on fire because of a chemical reaction
- ANS: 3

Functional Safety Analysis

- Safety = absence of unreasonable risk
- What is functional safety analysis?
  - Identify high-risk situations
  - Lower risk to reasonable levels
- It has 3 steps:
  - Identify hazards in a passenger vehicle's Electric/Electronic (E/E) system that could cause physical injury or damage to a person's health
  - Evaluate the risk of the hazardous situation so that we know how much we need to lower the risk
  - Via systems engineering, prevent accidents from occurring by lowering risk to reasonable levels.

Identifying Hazards

- Three general causes for accidents
  - Human error
    - The National Highway and Transportation Safety Administration estimates that 94% of car accidents in USA involve human error.
  - Technology error
    - Software bugs are a major source of technology error.
  - Human-technology interaction error
    - Whenever a human and a machine share control of a system, extra care needs to be taken when evaluating safety; the boundary between what the human is supposed to do and what the machine is supposed to do needs to be clear.
    - Drivers need to understand the warning signals coming from ADAS systems; interfaces need to make it clear when the vehicle expects us to take over.
    - Some companies (e.g., Waymo) has argued that low-levels of automation are not safe, since it is unrealistic to expect drivers to keep their hands on the steering wheel and be prepared to take over for long durations of time.
    - High levels of automation (L4/L5) will remove hazards due to human-technology interactions.

Quiz

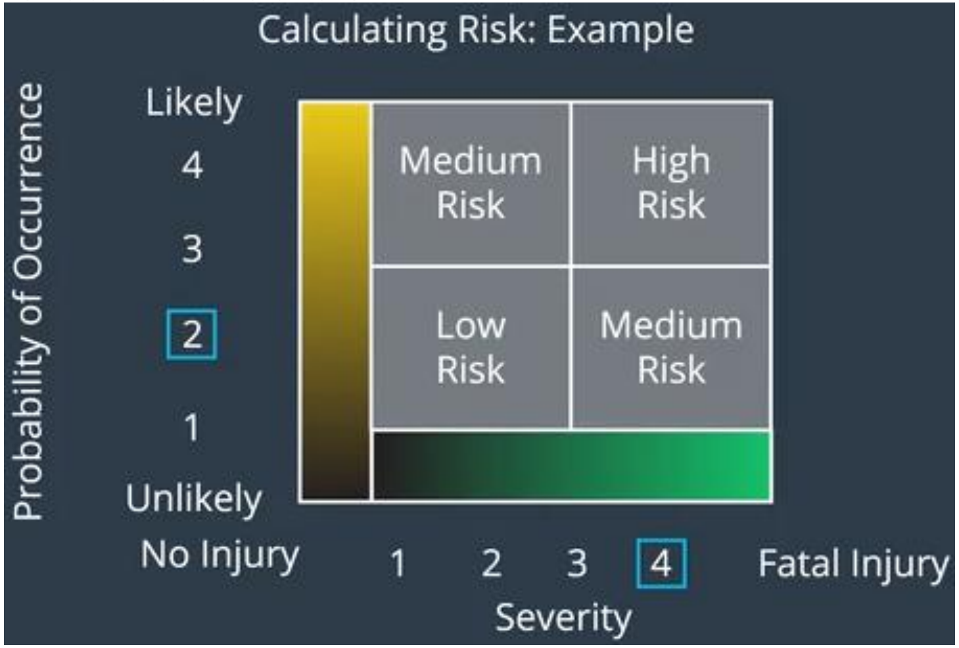
- Classify the causes for the following
  - 1. A software bug causes the vehicle to accelerate inadvertently.
  - 2. A driver goes too fast around a curve and falls off the road.
  - 3. A radar-based vehicle detection system is too sensitive and buzzes even when a crash is not imminent. The driver ignores warnings, which could lead to an accident if a crash were imminent.
- ANS: 1. Technology error; 2. Human error; 3. Human-technology interaction error

ASIL (Automotive Safety Integrity Level)

- The ISO 26262 standard defines 4 ASILs: ASIL A, ASIL B, ASIL C and ASIL D.
  - ASIL D represents a hazardous situation with the highest risk whereas ASIL A represents lower risk. ASIL is a key term in automotive functional safety.
  - There is one more level of risk below ASIL A called QM (Quality Managed), which means that development according to accepted quality principles is sufficient to reduce risk. QM is not covered by ISO 26262; it is covered by another standard IATF 16949.
- ASILs are determined by Hazard Analysis and Risk Assessment (HARA), discussed later.

Evaluating Risk

- risk = probability of occurrence × severity of the harm
- Probability of occurrence refers to how often a driving situation occurs.
  - Low for driving on a snowy mountainside (2); high for city driving (4).
- Severity of the harm refers to how bad the injury is.
  - A Power Steering malfunction may cause fatal injury for driving on a snowy mountainside (4), but unlikely or no injury for city driving (2)
- Risk of Power Steering malfunction for driving on a snowy mountainside is calculated as $2 \times 4 = 8$.

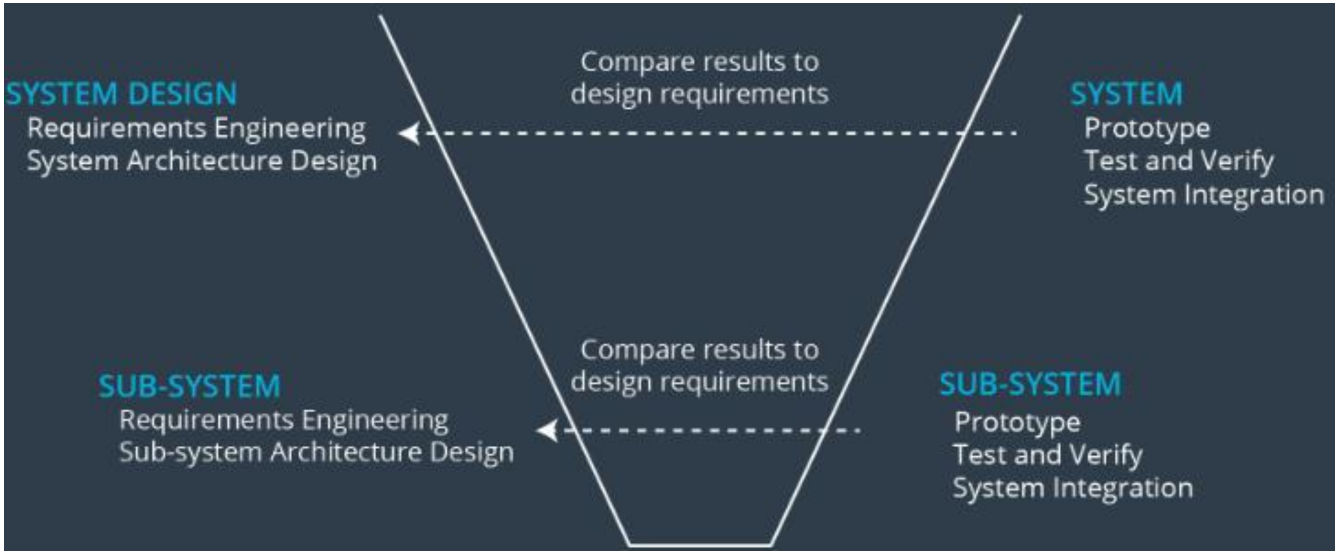


Quiz

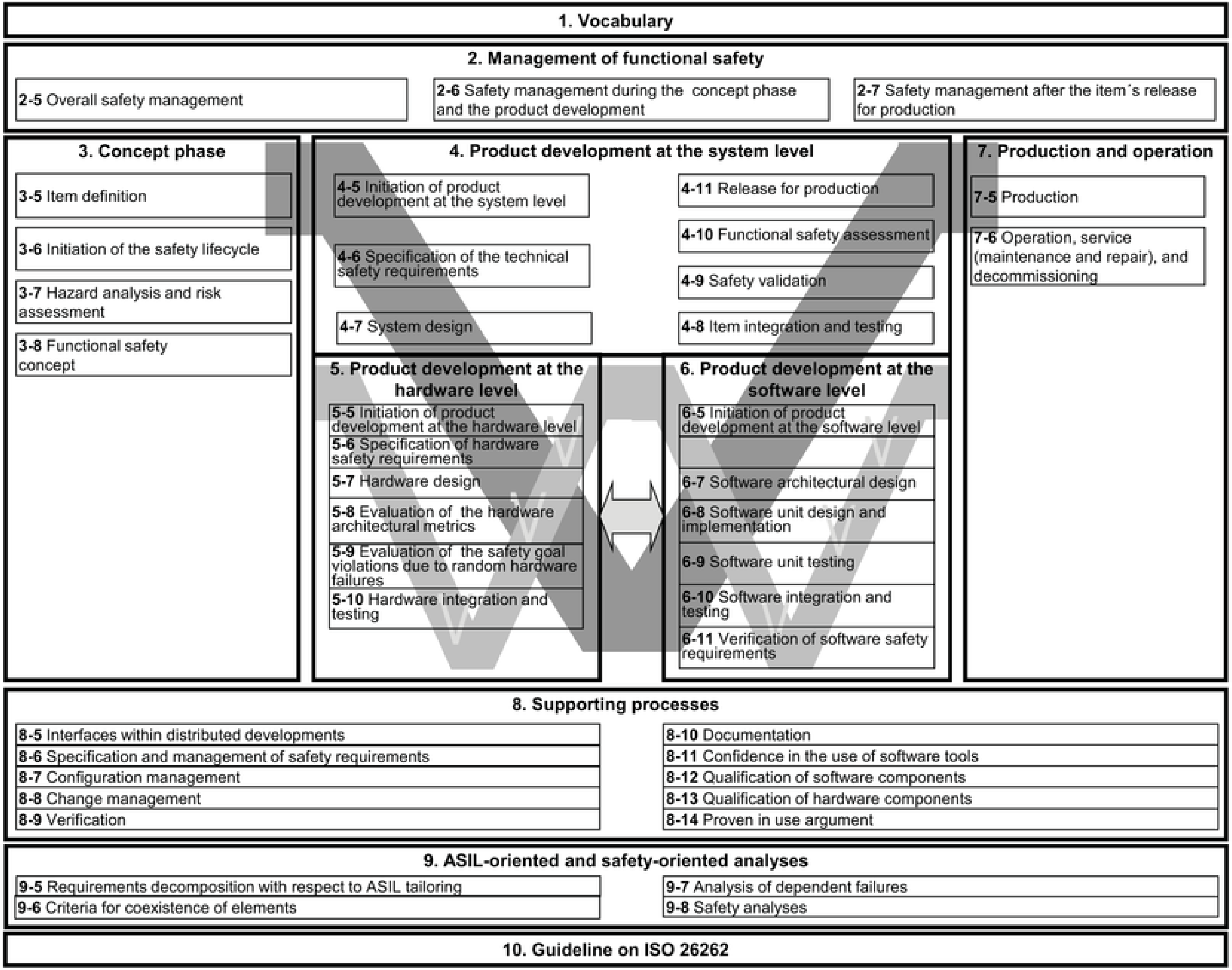
- Probability of occurrence:
  - 1 - Vehicle jumpstart needed
  - 2 - Driving in snow
  - 3 - Driving at night with no street lamps
  - 4 - Driving on the highway
- Severity:
  - 1 - No injury
  - 2 - Light and moderate injury
  - 3 - Severe and life-threatening injuries with probable survival
  - 4 - Life threatening injuries, survival uncertain, or fatal injury
- Calculate the risks for the following
  - A While driving on the highway, brakes fail at high-speed.
  - B Your vehicle requires a jumpstart. The power steering is not working but the vehicle is parked on the side of the road and not moving.
  - C While driving in the snow in the city, anti-lock brakes fail while driving at low speed.
  - D While driving at night on a street with no street lamps, the car headlamps fail.
- ANS: (Note: evaluations of severity may be subjective)
  - A:$4 \times 4 = 16$
  - B:$1 \times 1 = 1$
  - C:$2 \times 2 = 4$
  - D:$3 \times 2 = 6$

Reducing Risk with Systems Engineering

- The Systems Engineering V-Model
  - 1. Requirements engineering
  - 2. System architecture design and allocating requirements
  - 3. Testing and verification
  - 4. System Integration
- The top part of the V represents the entire vehicle as a single system. As you traverse down the left side of the V, you focus on smaller and smaller subsystems. As you go up the right side of the V, you integrate your subsystems into larger and larger systems.
- At the top left, you start with a bird's eye view of your entire vehicle. As you move down the left side, you start to split your vehicle into subsystems like climate control, entertainment system, steering system, braking system, etc. You then define requirements and system architectures for each sub-system.
- Then you focus on a sub-system, like the climate control system, and break down the climate control system into its own sub-systems like the ECU, the temperature sensor, the fan, the air filter, etc. Each sub-system will have its own design requirements and design architecture.
- You can connect the right side of the V and the left side of the V. For every prototype you make or test you run, you can check back to see if the results match the design specifications.



ISO 26262 Standard Document



1. Vocabulary

2. Management of functional safety

2-5 Overall safety management

2-6 Safety management during the concept phase and the product development

2-7 Safety management after the item´s release for production

3. Concept phase

3-5 Item definition

3-6 Initiation of the safety lifecycle

3-7 Hazard analysis and risk assessment

3-8 Functional safety concept

4. Product development at the system level

4-5 Initiation of product development at the system level

4-6 Specification of the technical safety requirements

4-7 System design

4-8 Item integration and testing

4-9 Safety validation

4-10 Functional safety assessment

4-11 Release for production

5. Product development at the hardware level

5-5 Initiation of product development at the hardware level

5-6 Specification of hardware safety requirements

5-7 Hardware design

5-8 Evaluation of the hardware architectural metrics

5-9 Evaluation of the safety goal violations due to random hardware failures

5-10 Hardware integration and testing

6. Product development at the software level

6-5 Initiation of product development at the software level

6-7 Software architectural design

6-8 Software unit design and implementation

6-9 Software unit testing

6-10 Software integration and testing

6-11 Verification of software safety requirements

7. Production and operation

7-5 Production

7-6 Operation, service (maintenance and repair), and decommissioning

8. Supporting processes

8-5 Interfaces within distributed developments

8-6 Specification and management of safety requirements

8-7 Configuration management

8-8 Change management

8-9 Verification

8-10 Documentation

8-11 Confidence in the use of software tools

8-12 Qualification of software components

8-13 Qualification of hardware components

8-14 Proven in use argument

9. ASIL-oriented and safety-oriented analyses

9-5 Requirements decomposition with respect to ASIL tailoring

9-6 Criteria for coexistence of elements

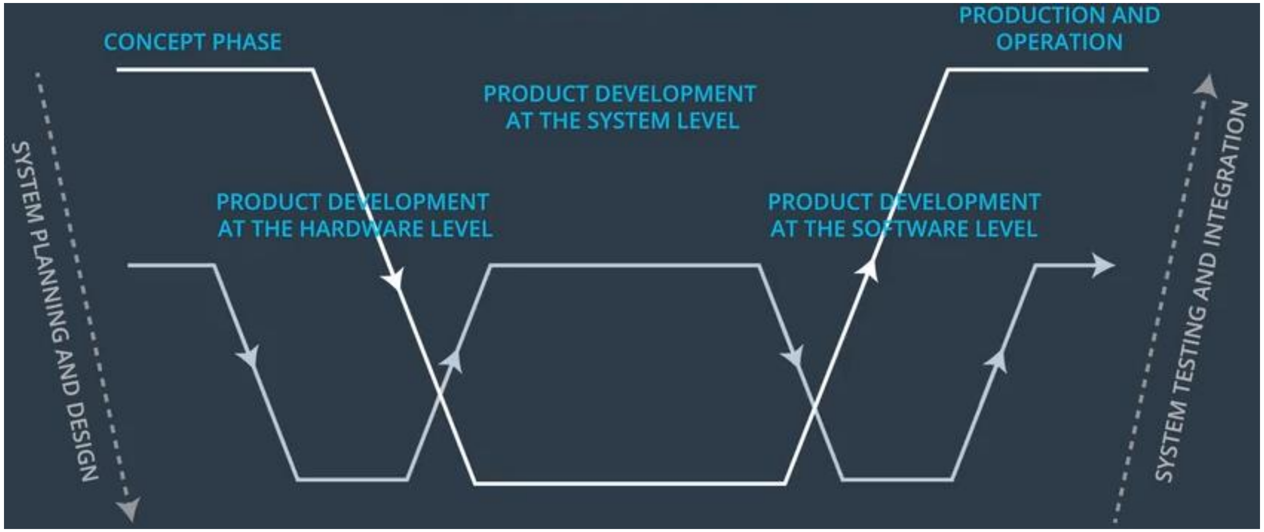
9-7 Analysis of dependent failures

9-8 Safety analyses

10. Guideline on ISO 26262

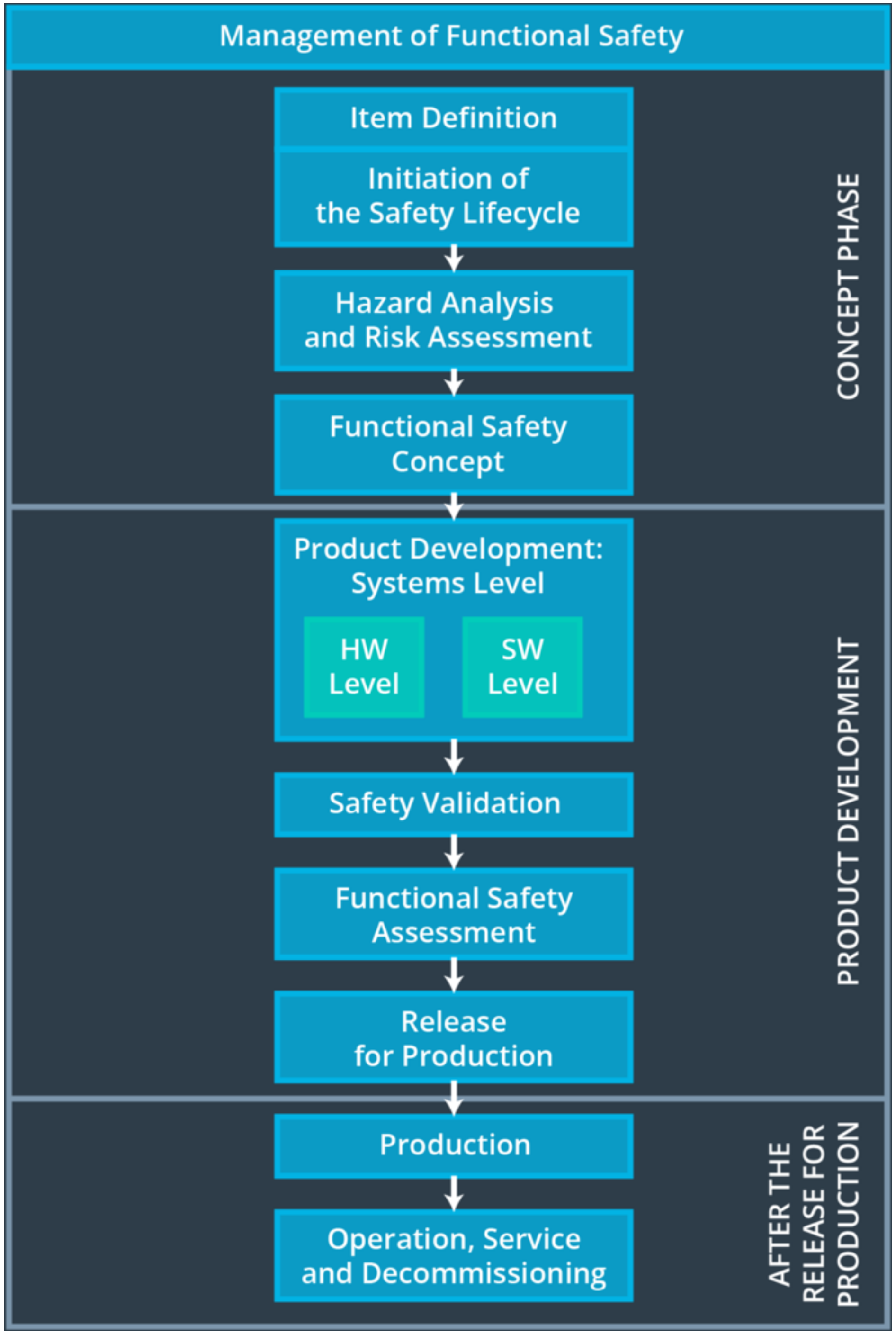
ISO 26262 V Model

- ISO 26262 compliance is not legally required, but highly desirable.
- Higher up in the V model represents integrated systems
  - The big basin (“flat-down-flat-up-flat” line).
- Lower down in the V model represents subsystems
  - Two small basins (the “flat-down-flat-up-flat-down-flat-up-flat” line)
  - Left small basin for hardware development; Right small basin for software development;
- Downward arrows on the left side of each basin represent specifying requirements and designing a system architecture.
- Upward arrows on the right side of each basin represent prototyping, testing and integration.

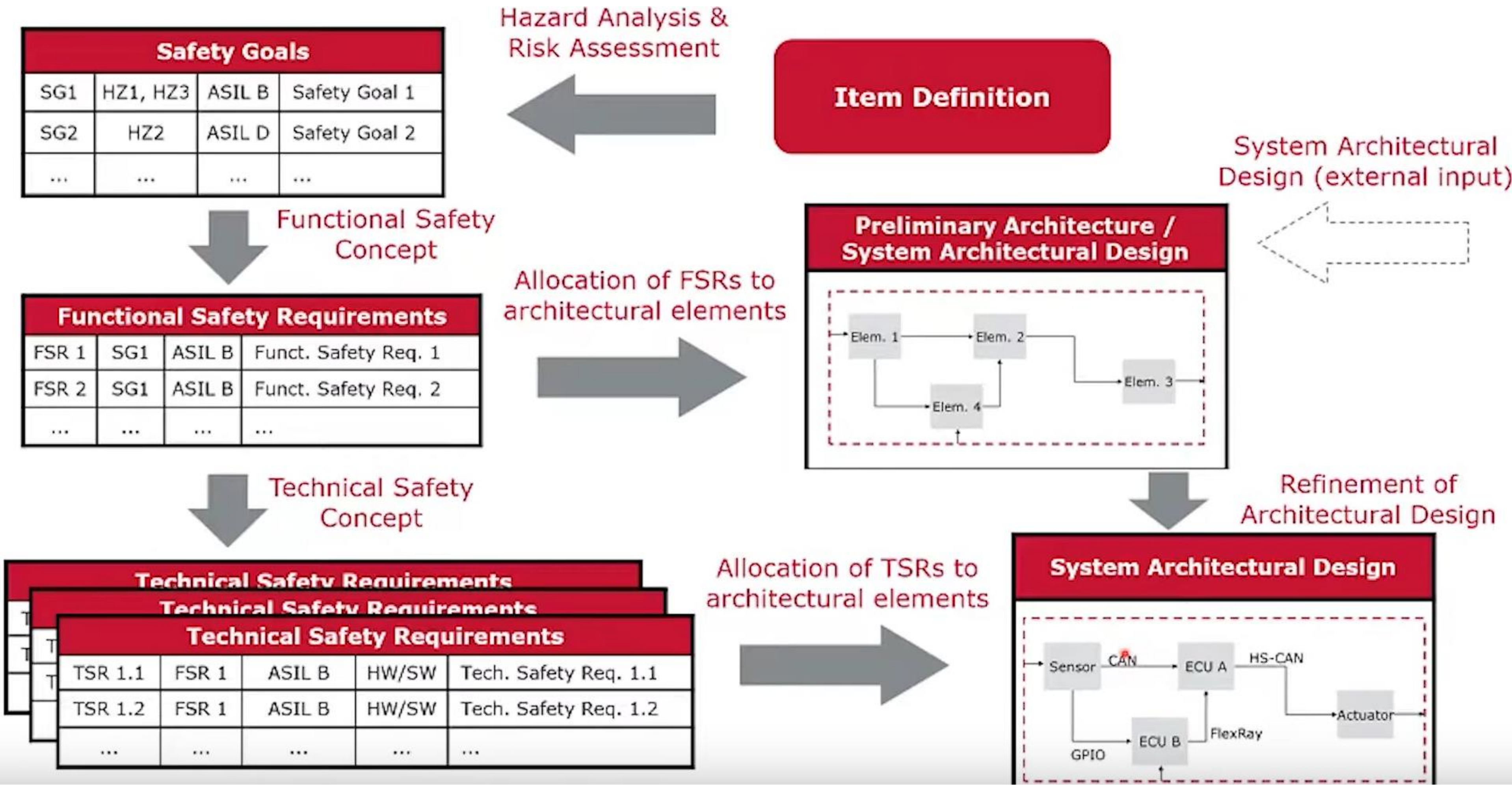

CONCEPT PHASE
PRODUCTION AND OPERATION
PRODUCT DEVELOPMENT AT THE SYSTEM LEVEL
PRODUCT DEVELOPMENT AT THE HARDWARE LEVEL
PRODUCT DEVELOPMENT AT THE SOFTWARE LEVEL
SYSTEM PLANNING AND DESIGN
SYSTEM TESTING AND INTEGRATION

Flattened V Model

- You can flatten out the V model to see it from a more linear perspective. The image gives the major steps involved in a functional safety project.


Management of Functional Safety
Item Definition
Initiation of the Safety Lifecycle
Hazard Analysis and Risk Assessment
Functional Safety Concept
CONCEPT PHASE
Product Development: Systems Level
HW Level
SW Level
Safety Validation
Functional Safety Assessment
Release for Production
PRODUCT DEVELOPMENT
Production
Operation, Service and Decommissioning
AFTER THE RELEASE FOR PRODUCTION

ISO 26262 Development Process



Item Definition

Hazard Analysis & Risk Assessment

Safety Goals

SG1	HZ1, HZ3	ASIL B	Safety Goal 1
SG2	HZ2	ASIL D	Safety Goal 2
...	...	...	...

Functional Safety Concept

Functional Safety Requirements

FSR 1	SG1	ASIL B	Funct. Safety Req. 1
FSR 2	SG1	ASIL B	Funct. Safety Req. 2
...	...	...	...

Allocation of FSRs to architectural elements

System Architectural Design (external input)

Preliminary Architecture / System Architectural Design

Technical Safety Concept

Technical Safety Requirements

TSR 1.1	FSR 1	ASIL B	HW/SW	Tech. Safety Req. 1.1
TSR 1.2	FSR 1	ASIL B	HW/SW	Tech. Safety Req. 1.2
...	...	...	...	...

Refinement of Architectural Design

Allocation of TSRs to architectural elements

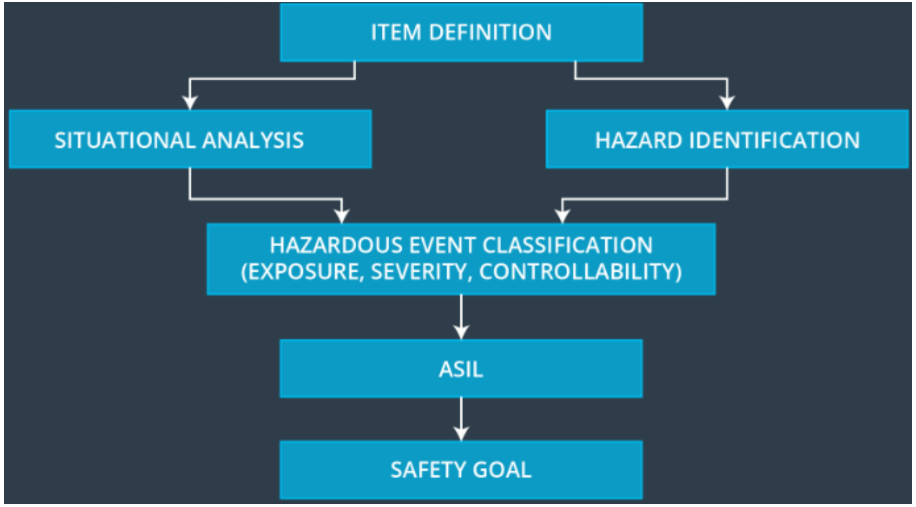
System Architectural Design

Outline

- Introduction to functional safety
- Hazard Analysis and Risk Assessment (HARA)
- Functional Safety Concept
- Technical Safety Concept
- Functional Safety for Hardware and Software

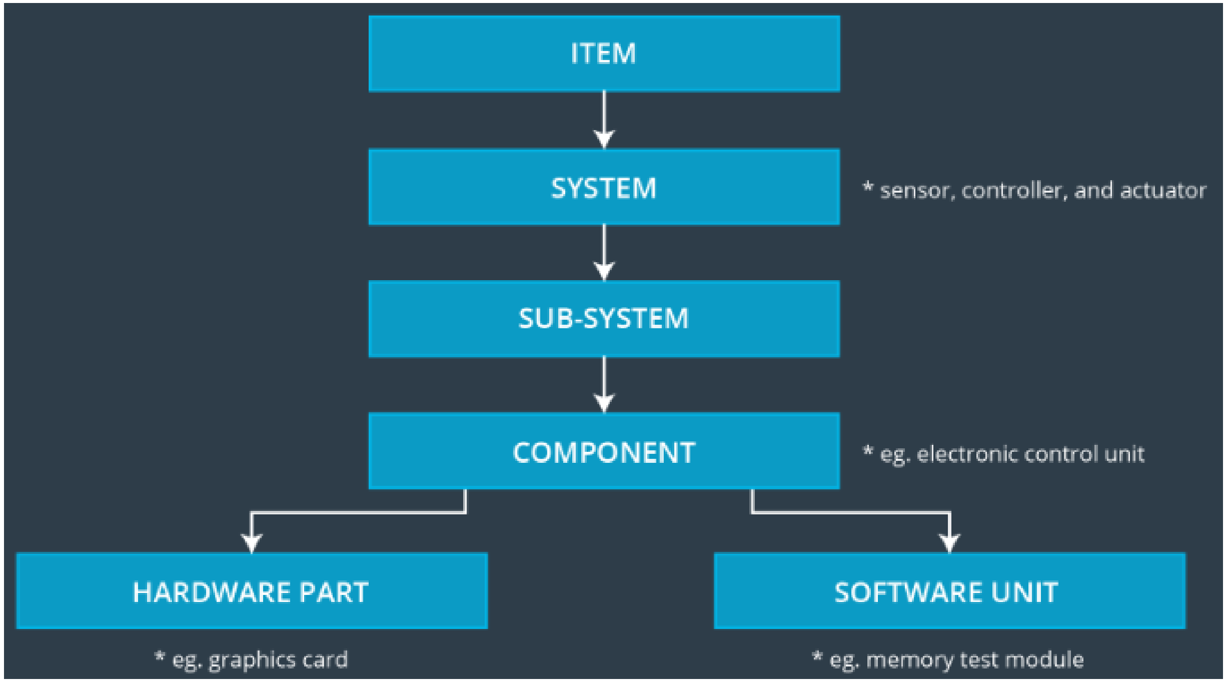
Hazard Analysis and Risk Assessment (HARA)

- This is where we brainstorm to imagine hazards where the system malfunctions and causes injury or harm. Then the risk is calculated for each hazardous scenario.
- Item Definition describes which vehicle system is under consideration. Followed by 5 steps:
- Situational Analysis
  - Choose different driving scenarios like driving on a bumpy road, being towed, and driving on the freeway.
- Hazard Identification
  - Figure out how the system could malfunction, e.g., an electronic parking brake failure.
- Hazardous Event Classification According to Exposure, Severity and Controllability
  - Combine situations and hazards together, i.e., take a malfunction and then think about the malfunction under different driving scenarios. Like if the electronic parking brake failed while the vehicle was parked on a steep hill.
  - Then calculate three metrics called exposure, severity and controllability, which will depend on the hazard, the driving scenario and what might happen when the hazard occurs under the scenario.
- ASIL Determination
  - After you have calculated exposure, severity and controllability, you can now determine the ASIL based on a table.
- Safety Goal
  - A safety goal is a type of engineering requirement for vehicle functional safety; for example, "The electronic parking brake system shall always be engaged when the vehicle is in park on a gradient that is greater than 10 degrees".



ISO 26262 Definition of System Hierarchy

- An Item is a high-level system in the vehicle
- A system is a set of elements that have at least a sensor, controller and an actuator.
  - The LAS has two sensors: a camera sensor and a driver steering torque sensor. The ECUs are the controllers, and the actuator is the motor that turns the steering wheel.
- A sub-system is a smaller piece of a system.
- A component may be an ECU, which consists of hardware parts (CPU, GPU…) and software units (application program, memory test…)

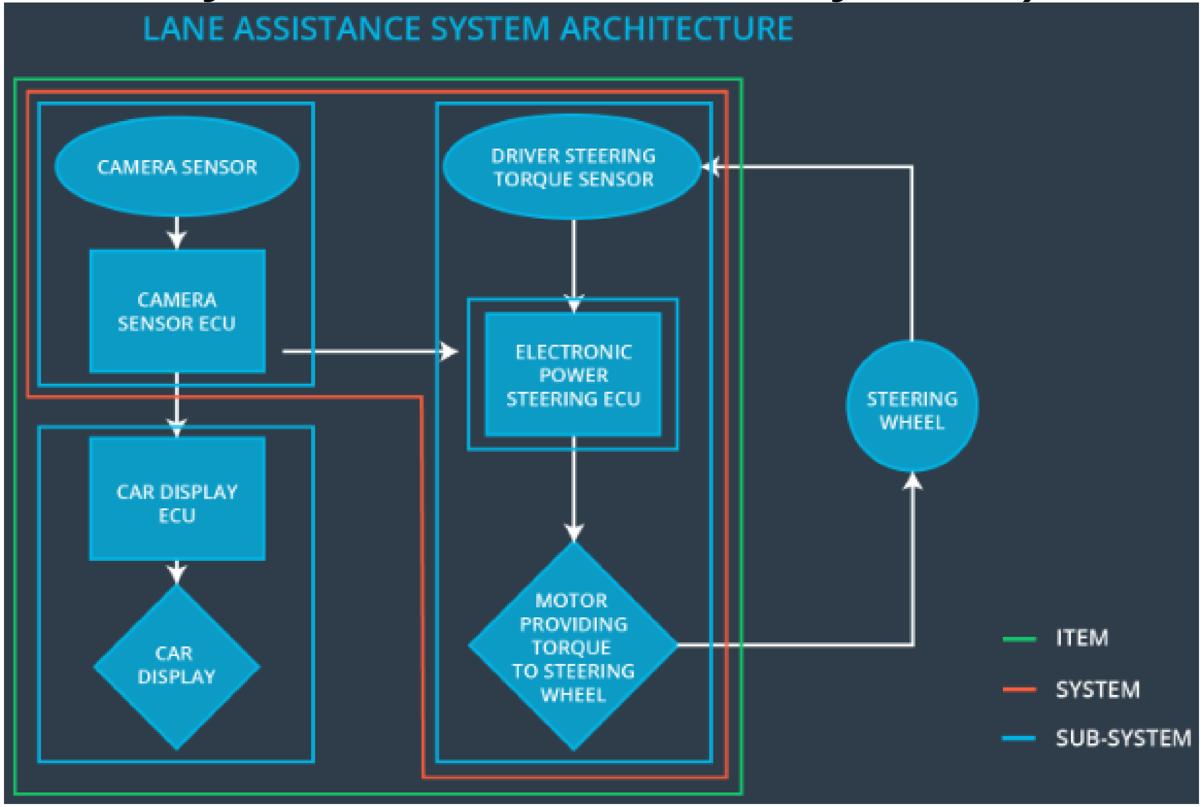


Running Example: Lane Assistance System (LAS)

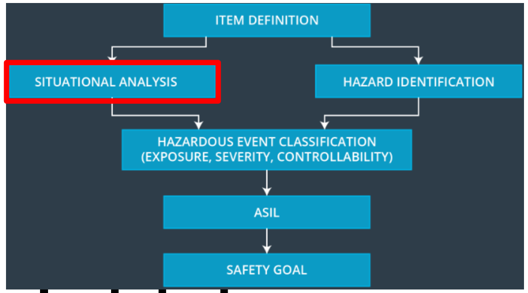
- A LAS has two functions:
  - Lane Departure Warning (LDW)
  - Lane Keeping Assistance (LKA)
- If a driver departs a lane without using a turn signal, the system assumes that the driver has become distracted and did not mean to leave the lane. Two things will happen:
  - The LDW function will vibrate the steering wheel to provide a driver a haptic feedback.
  - The LKA function will move the steering wheel so that the wheels turn towards the center of the lane in order to stay in ego lane.
- You can assume that the engineering requirement came from a product engineering team, and the safety engineer’s job will be to add extra requirements to ensure functional safety.

LAS Architecture

- LAS consists of 1 item (green box), 1 system (red box), and 4 subsystems (blue boxes).
- Subsystems may be nested (Electronic Power Steering ECU is a subsystem within a subsystem).


LANE ASSISTANCE SYSTEM ARCHITECTURE
CAMERA SENSOR
CAMERA SENSOR ECU
CAR DISPLAY ECU
CAR DISPLAY
DRIVER STEERING TORQUE SENSOR
ELECTRONIC POWER STEERING ECU
MOTOR PROVIDING TORQUE TO STEERING WHEEL
STEERING WHEEL
🟩 ITEM
🟥 SYSTEM
🟦 SUB-SYSTEM

Situational Analysis



ITEM DEFINITION

SITUATIONAL ANALYSIS

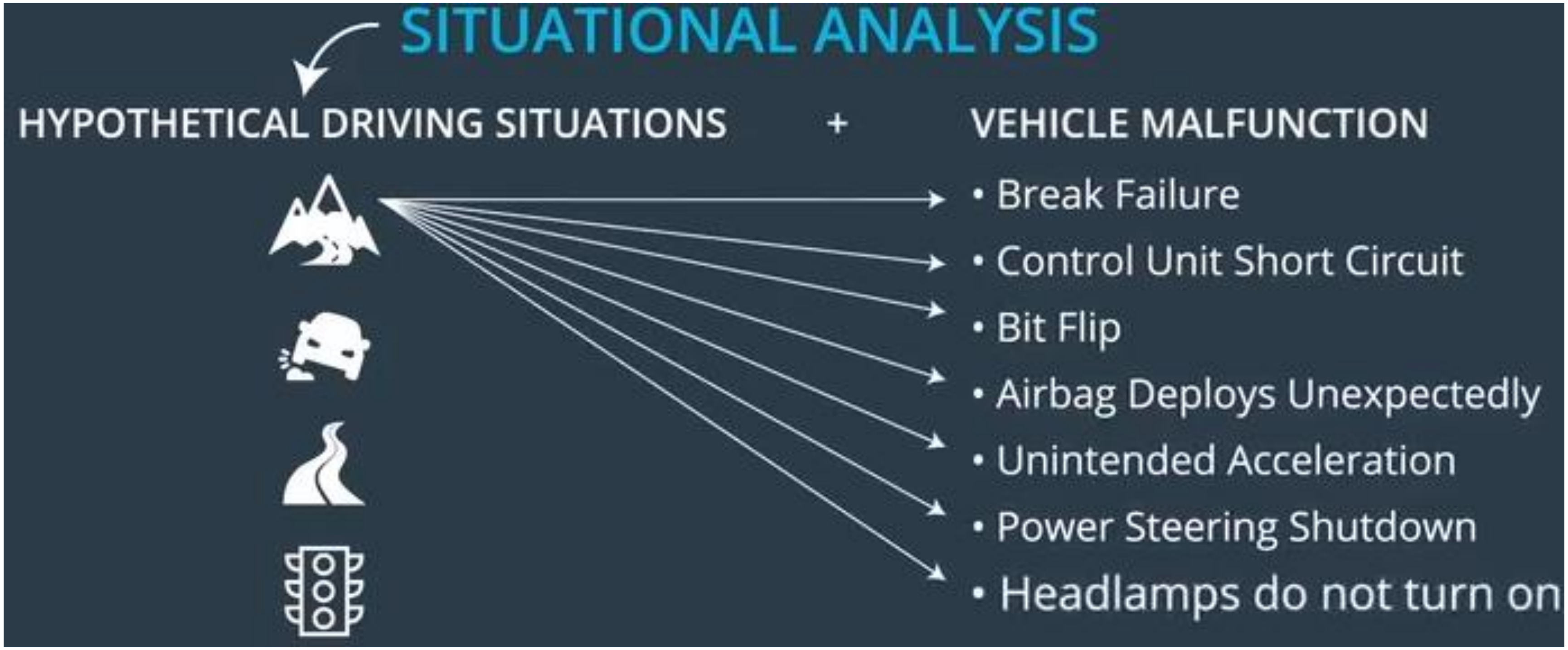
HAZARD IDENTIFICATION

HAZARDOUS EVENT CLASSIFICATION (EXPOSURE, SEVERITY, CONTROLLABILITY)

ASIL

SAFETY GOAL

- Brainstorming about hypothetical driving situations and analyze what would happen in case of vehicle malfunction.



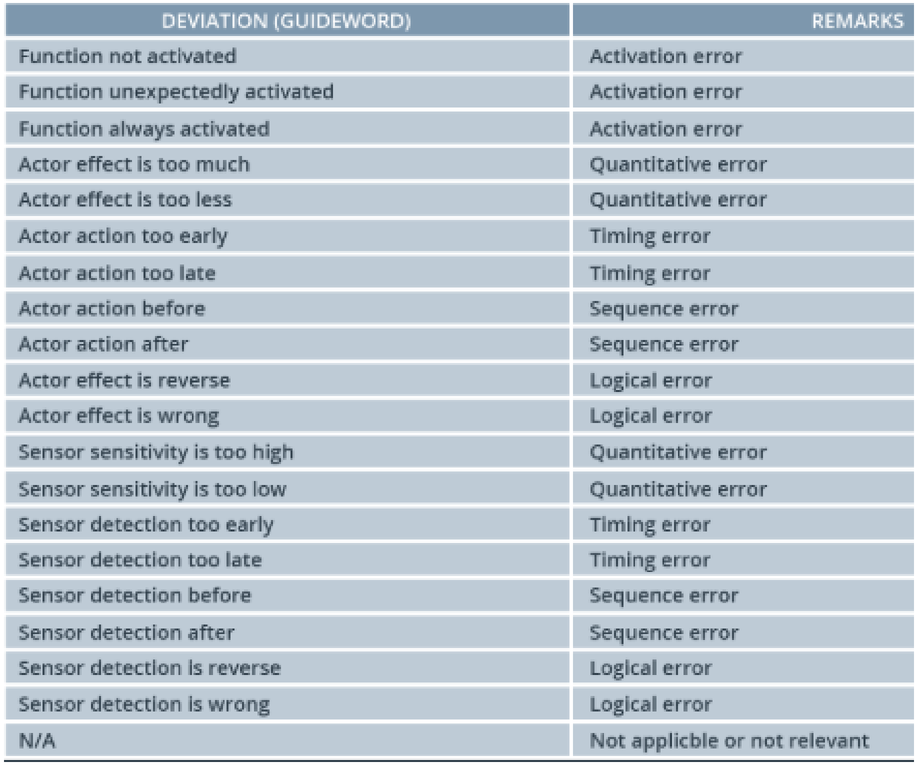
SITUATIONAL ANALYSIS

HYPOTHETICAL DRIVING SITUATIONS + VEHICLE MALFUNCTION

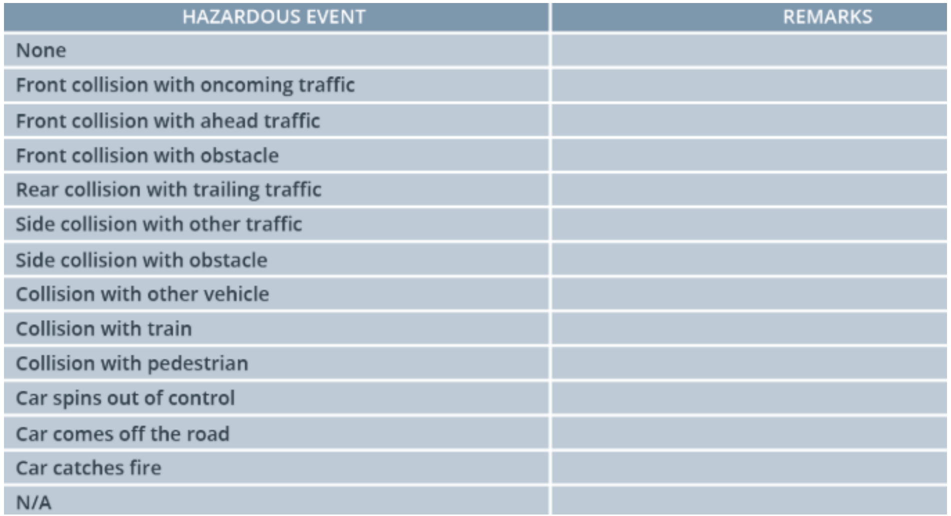
- Break Failure
- Control Unit Short Circuit
- Bit Flip
- Airbag Deploys Unexpectedly
- Unintended Acceleration
- Power Steering Shutdown
- Headlamps do not turn on

Predefined Guidewords

- Use predefined guidewords to help unify terminology
  - Below shows guidewords for deviation (malfunction) and Hazardous Event
- Example:
  - [Operational Mode] on [Operational Scenario] during [Environmental Details] with [Situational Details] and [Item Usage] system. Terms in brackets should be taken from standard guideword tables.
  - Ex1. [backward driving] on [city road] during [fog] with [low speed] and [correctly used] system.
  - Ex2 (LAS example): [Normal driving] on a [highway] during [rain (slippery road)] with [high speed] and [correctly used] system

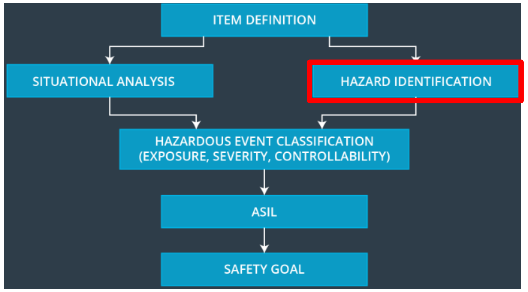


DEVIATION (GUIDEWORD)	REMARKS
Function not activated	Activation error
Function unexpectedly activated	Activation error
Function always activated	Activation error
Actor effect is too much	Quantitative error
Actor effect is too less	Quantitative error
Actor action too early	Timing error
Actor action too late	Timing error
Actor action before	Sequence error
Actor action after	Sequence error
Actor effect is reverse	Logical error
Actor effect is wrong	Logical error
Sensor sensitivity is too high	Quantitative error
Sensor sensitivity is too low	Quantitative error
Sensor detection too early	Timing error
Sensor detection too late	Timing error
Sensor detection before	Sequence error
Sensor detection after	Sequence error
Sensor detection is reverse	Logical error
Sensor detection is wrong	Logical error
N/A	Not applicble or not relevant



HAZARDOUS EVENT	REMARKS
None	
Front collision with oncoming traffic	
Front collision with ahead traffic	
Front collision with obstacle	
Rear collision with trailing traffic	
Side collision with other traffic	
Side collision with obstacle	
Collision with other vehicle	
Collision with train	
Collision with pedestrian	
Car spins out of control	
Car comes off the road	
Car catches fire	
N/A	

Hazard Identification for LDW



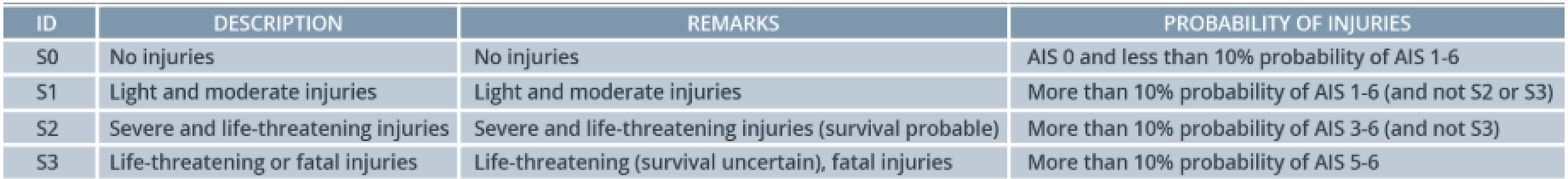
Function	LDW function shall apply an oscillating steering torque to provide the driver haptic feedback.
Malfunction	Actor effect is too much.
Malfunction details	LDW function applies an oscillating torque with very high torque (above limit).
Hazardous event	Collision with other vehicle.
Event details	High haptic feedback can affect driver's ability to steer as intended. The driver could lose control of the vehicle and collide with another vehicle or with road infrastructure.
Summary description	The LDW function applies too high an oscillating torque to the steering wheel (above limit).

Hazard Identification for LKA

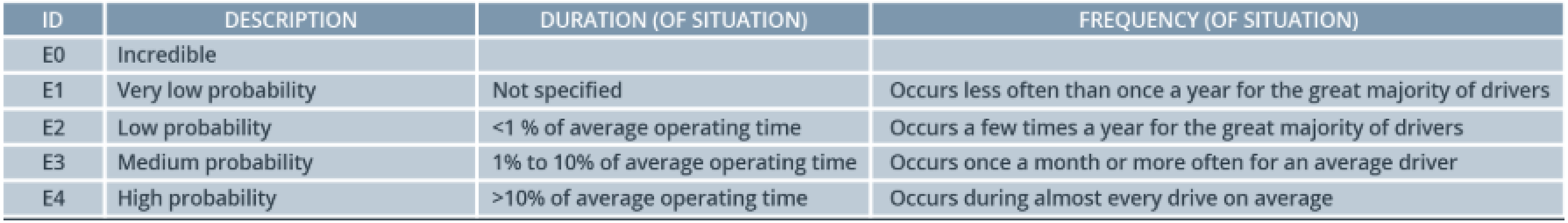
Function	LKA function will move the steering wheel so that the wheels turn towards the center of the lane.
Malfunction	Driver takes both hands off the wheel.
Malfunction details	Driver misuses the function by taking both hands off the wheel and incorrectly treating the car as an AV, causing LKA function to be always activated
Hazardous event	Collision with other vehicle.
Event details	…
Summary description	…

ISO 26262 Risk Measurement

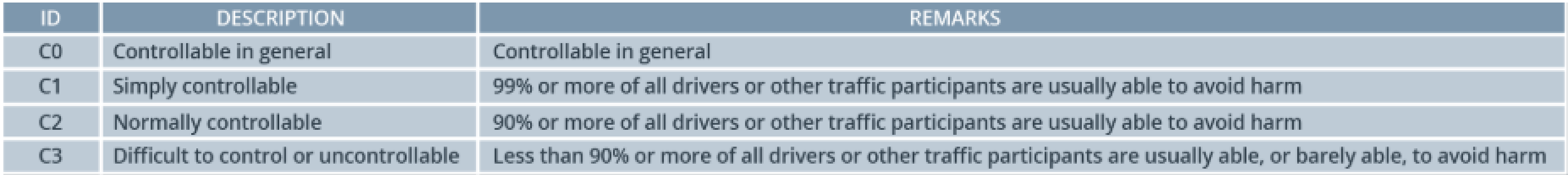
- Recall previous definition: risk = probability of occurrence × severity of the harm
- ISO 26262 refines risk definition to: risk = severity × exposure × controllability
  - Severity is the same: exposure corresponds to probability of occurrence; controllability measures how likely driver could gain control of the vehicle
  - Probability of loss = exposure × controllability
- From top to bottom: table for determining severity, exposure, controllability
  - See notes for definitions of AIS (accident injury scale) for determining severity.



ID	DESCRIPTION	REMARKS	PROBABILITY OF INJURIES
S0	No injuries	No injuries	AIS 0 and less than 10% probability of AIS 1-6
S1	Light and moderate injuries	Light and moderate injuries	More than 10% probability of AIS 1-6 (and not S2 or S3)
S2	Severe and life-threatening injuries	Severe and life-threatening injuries (survival probable)	More than 10% probability of AIS 3-6 (and not S3)
S3	Life-threatening or fatal injuries	Life-threatening (survival uncertain), fatal injuries	More than 10% probability of AIS 5-6



ID	DESCRIPTION	DURATION (OF SITUATION)	FREQUENCY (OF SITUATION)
E0	Incredible		
E1	Very low probability	Not specified	Occurs less often than once a year for the great majority of drivers
E2	Low probability	<1 % of average operating time	Occurs a few times a year for the great majority of drivers
E3	Medium probability	1% to 10% of average operating time	Occurs once a month or more often for an average driver
E4	High probability	>10% of average operating time	Occurs during almost every drive on average

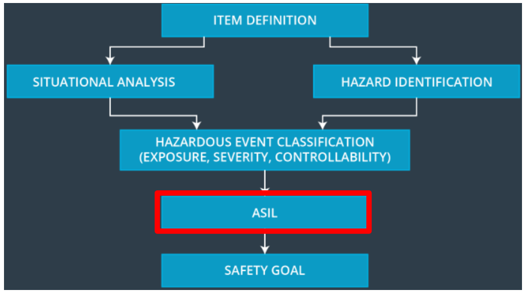


ID	DESCRIPTION	REMARKS
C0	Controllable in general	Controllable in general
C1	Simply controllable	99% or more of all drivers or other traffic participants are usually able to avoid harm
C2	Normally controllable	90% or more of all drivers or other traffic participants are usually able to avoid harm
C3	Difficult to control or uncontrollable	Less than 90% or more of all drivers or other traffic participants are usually able, or barely able, to avoid harm

LAS Example

- Situation: [Normal driving] on a [highway] during [rain (slippery road)] with [high speed] and [correctly used] system.
  - Severity: S3 (life-threatening or fatal injuries)
    - Highway driving implies high severity of harm.
  - Exposure: E3 (medium probability)
    - The situation is not uncommon, but also not everyday.
  - Controllability: C3 (difficult to control or uncontrollable)
    - Drive may lose control if the steering wheel vibration is excessive.

ASIL Determination



ITEM DEFINITION

SITUATIONAL ANALYSIS

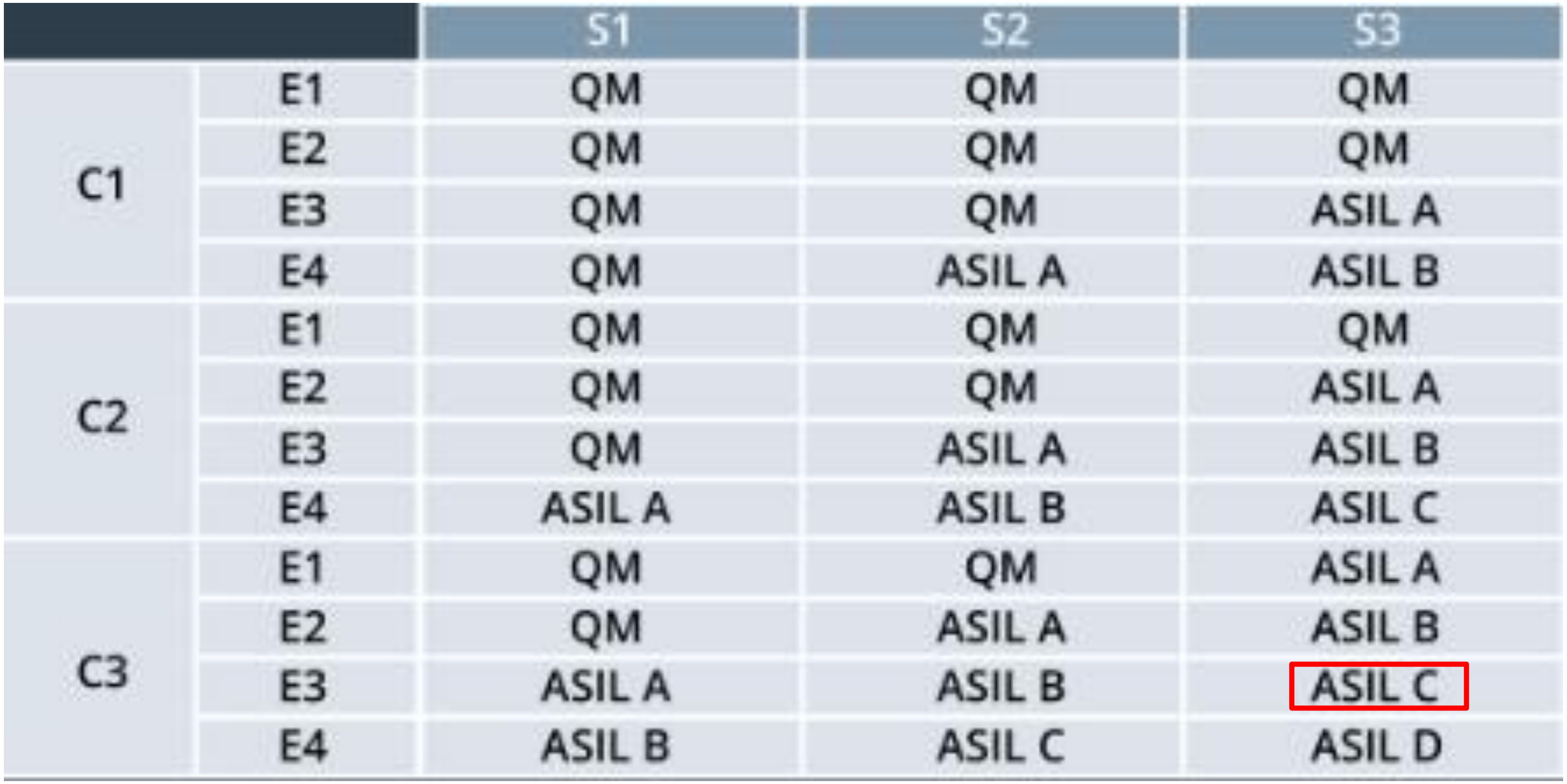
HAZARD IDENTIFICATION

HAZARDOUS EVENT CLASSIFICATION (EXPOSURE, SEVERITY, CONTROLLABILITY)

ASIL

SAFETY GOAL

- LDW is determined to be ASIL C (S3 × E3 × C3).



		S1	S2	S3
C1	E1	QM	QM	QM
	E2	QM	QM	QM
	E3	QM	QM	ASIL A
	E4	QM	ASIL A	ASIL B
C2	E1	QM	QM	QM
	E2	QM	QM	ASIL A
	E3	QM	ASIL A	ASIL B
	E4	ASIL A	ASIL B	ASIL C
C3	E1	QM	QM	ASIL A
	E2	QM	ASIL A	ASIL B
	E3	ASIL A	ASIL B	ASIL C
	E4	ASIL B	ASIL C	ASIL D

LDW in Other Driving Situations

- Driving on a wet road on a city street at low speed. Now LDW function is ASIL A (S1 × E3 × C3):
  - Severity: S1 (light and moderate injuries due to low speed collision)
  - Exposure: E3 (medium probability)
  - Controllability: C3 (excessive vibration of steering wheel makes it difficult to control even at low speeds)
  - The driver is driving more slowly with all other conditions remaining the same, so the risk has gone down.
- Driving on a dry road on a city street at low speed. Now LDW function is ASIL B (S1 × E4 × C3):
  - Severity: S1
  - Exposure: E4 (high probability assuming no rain on most days)
  - Controllability: C3
  - The increased ASIL (from A to B) is due to higher exposure: A driver is more likely to be driving on a dry road than a wet road, so there is a higher probability that a random malfunction will occur on a dry road; hence risk increases for the dry road scenario.
- Driving on a dry road on a highway at high speed. Now LDW function is ASIL D (S3 × E4 × C3):
  - Severity: S3
  - Exposure: E4
  - Controllability: C3
- When more than one situation maps to the same hazard, choose the highest ASIL level
  - LDW function for city driving (regardless of wet or dry) is ASIL B.

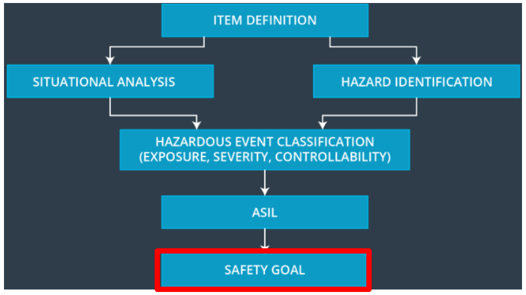
ASIL ≠ Importance

- A high-ASIL function is not necessarily more “important” than a low-ASIL function. And “Importance” is not a formally defined concept in ISO 26262.
- Consider two safety functions SF1 and SF2:
  - SF1 is ASIL C (S2 × E4 × C3)
  - SF2 is ASIL B (S3 × E2 × C3)
  - SF2 is assigned a lower ASIL due to its low exposure (E2), but its severity (S3) is higher than SF1’s (S2), hence malfunction of SF2 may cause more severe harm than malfunction of SF1.

Analysis and Testing for Different ASILs

- Higher ASIL levels require more analysis and testing to reduce risk to acceptable levels. A few examples of extra measures that need to be taken for higher ASILs:
  - Fault Tree Analysis (FTA) is mandatory for ASIL C and ASIL D.
  - ASIL D requires hardware failure rate of <10 FIT; ASIL B/C require hardware failure rate of <100 FIT; ASIL A requires hardware failure rate of <1000 FIT.
    - The FIT rate of a component is the number of failures expected in one billion hours of operation.
  - Software units with ASIL D require more rigorous testing like MC/DC coverage whereas ASIL B only mandates DC coverage.

Safety Goal for LDW

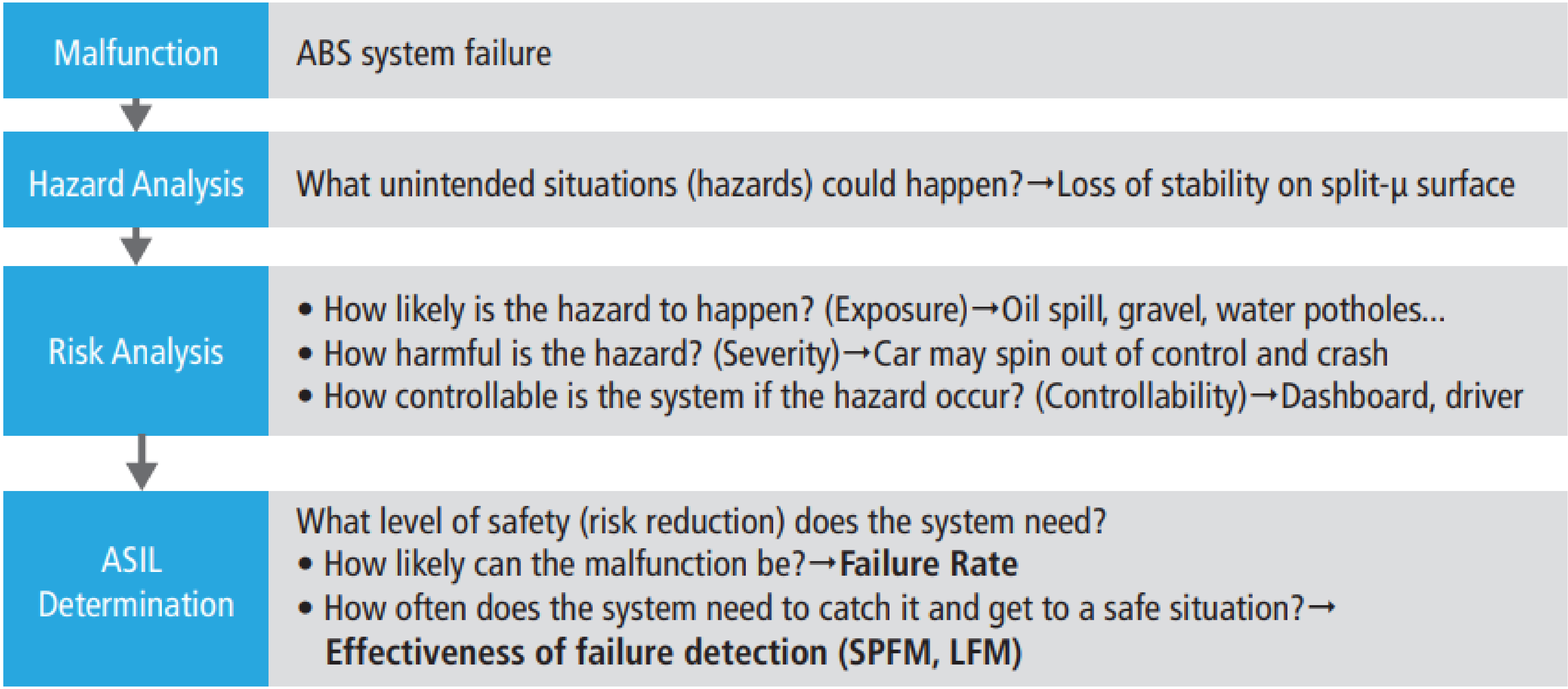


- For the hazardous event:
  - “LDW function applies too high an oscillating torque to the steering wheel.”
- Safety goal:
  - “The oscillating torque from the LDW function shall be limited.”

Quiz: Safety Goal for LKA

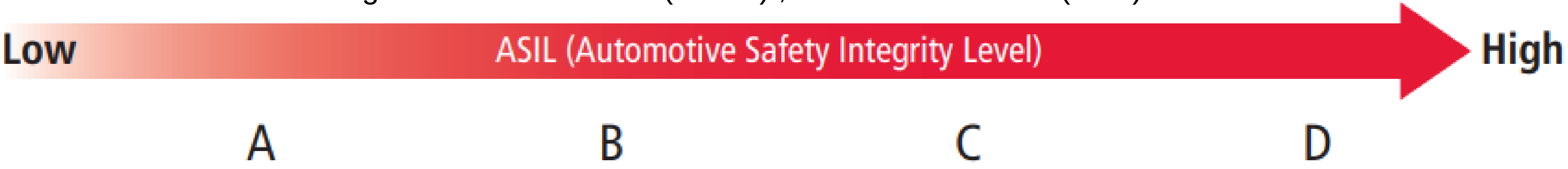
- What is the safety goal for LKA function?
  - – 1. It shall provide a low enough torque to the steering wheel such that the driver cannot feel the system working.
  - – 2. It shall provide torque to the steering wheel such that the vehicle moves away from the center of the lane.
  - – 3. It shall only turn on when it is raining.
  - – 4. It shall be time-limited and the additional steering torque shall end after a given time interval so that the driver cannot misuse the system for AD
- ANS: 4

HARA Summary



Malfunction	ABS system failure
Hazard Analysis	What unintended situations (hazards) could happen?→Loss of stability on split-µ surface
Risk Analysis	• How likely is the hazard to happen? (Exposure)→Oil spill, gravel, water potholes... • How harmful is the hazard? (Severity)→Car may spin out of control and crash • How controllable is the system if the hazard occur? (Controllability)→Dashboard, driver
ASIL Determination	What level of safety (risk reduction) does the system need? • How likely can the malfunction be?→**Failure Rate** • How often does the system need to catch it and get to a safe situation?→ **Effectiveness of failure detection (SPFM, LFM)**

Single-Point Fault Metric (SPFM) ; Latent Fault Metric (LFM)



https://www.cadence.com/content/dam/cadence-www/global/en_US/documents/solutions/automotive-functional-safety-wp.pdf

Outline

- Introduction to functional safety
- Hazard Analysis and Risk Assessment (HARA)
- Functional Safety Concept
- Technical Safety Concept
- Functional Safety for Hardware and Software

Functional Safety Concept

- In functional safety, "concept" is synonymous with "document".
- It includes the following steps:
  - Derive functional safety requirements.
  - Refine the item architecture.
  - Allocate functional safety requirements to the item architecture
  - Determine ASILs for the subsystems

Malfunctions

- Based on information from HARA, we identify 2 malfunctions for LDW function:
  - “The LDW function applies an oscillating torque with very high torque amplitude (above limit).”
  - “The LDW function applies an oscillating torque with very high torque frequency (above limit).”
- And 1 malfunction for LKA function:
  - “The LKA function is not limited in time duration which leads to misuse as an AD function.”

Formal Safety Analysis Methods

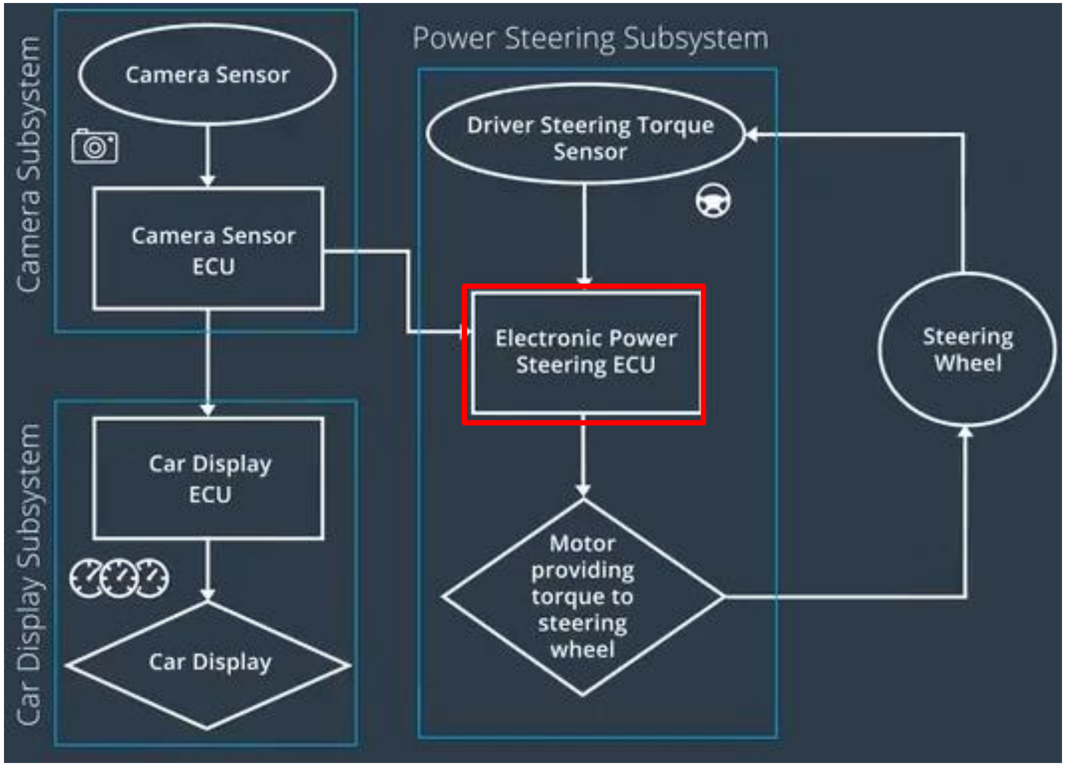
- There are formal methods for:
  - Deriving safety requirements
  - Identifying conditions and causes that could lead to requirement violations
  - Identifying other hazards not identified in the Hazard Analysis and Risk Assessment
- They include (not covered in this course):
  - Failure Modes and Effects Analysis (FMEA)
  - Hazard and Operability Study (HAZOPS)
  - Failure Modes Effects and Diagnostic Analysis (FMEDA)
  - Fault Tree Analysis (FTA)
  - Reliability Block Diagram
  - …
- But we will just use some basic logic to derive functional safety requirements from the safety goals.

Functional Safety Requirements

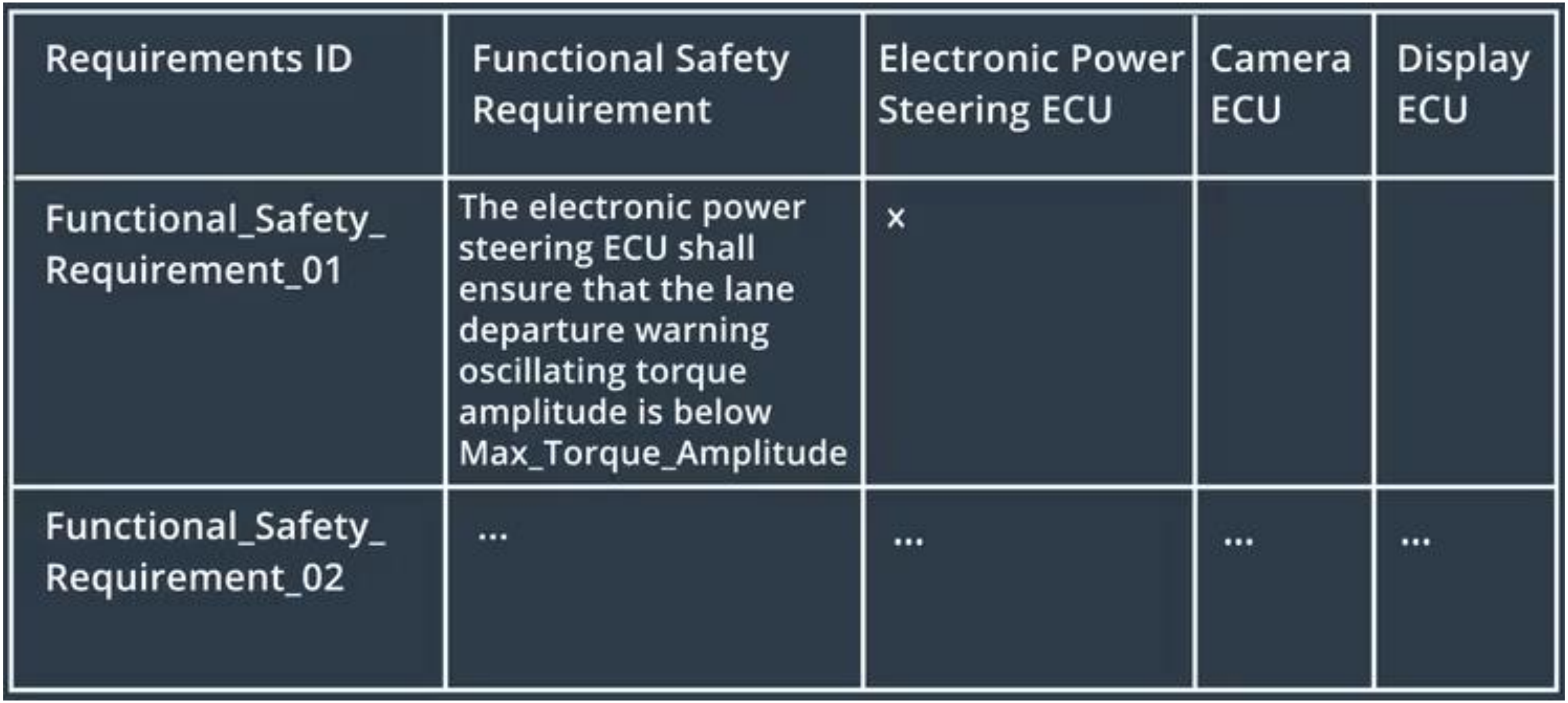
- Based on the 2 safety goals from HARA:
  - 1. “The oscillating steering torque from the LDW function shall be limited.”
  - 2. “The LKA function shall be time limited, and the additional steering torque shall end after a given time interval so that the driver cannot misuse the system for AD.”
- We derive functional safety requirements from safety goal 1 of the LDW function:
  - 1. “The lane assistance item shall ensure that the lane departure oscillating torque amplitude is below Max_Torque_Amplitude.”
  - 2. “The lane assistance item shall ensure that the lane departure oscillating torque frequency is below Max_Torque_Frequency.”
- And from safety goal 2 of the LKA function:
  - 3. “The lane assistance item shall ensure that the lane keeping assistance torque is applied for only Max_Duration”.

Allocating Requirements to Architecture

- We need to allocate these requirements to subsystems. If we allocate them to the Electronic Power Steering ECU, the requirements become:
  - 1. “The Electronic Power Steering ECU shall ensure that the lane departure oscillating torque amplitude is below Max_Torque_Amplitude.”
  - 2. “The Electronic Power Steering ECU shall ensure that the lane departure oscillating torque frequency is below Max_Torque_Frequency.”
  - 3. “The Electronic Power Steering ECU shall ensure that the lane keeping assistance torque is applied for only Max_Duration.”



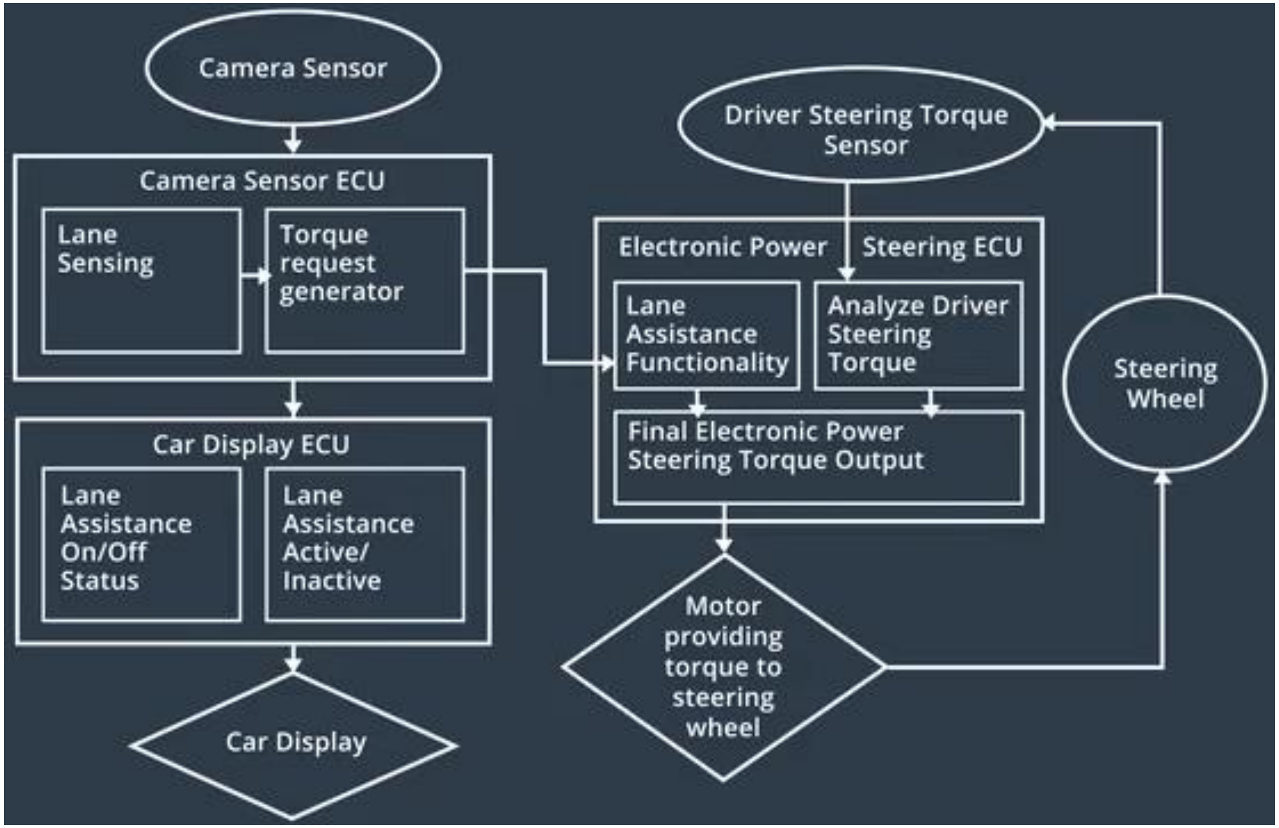
Requirements to Architecture Allocation Table



Requirements ID	Functional Safety Requirement	Electronic Power Steering ECU	Camera ECU	Display ECU
Functional_Safety_Requirement_01	The electronic power steering ECU shall ensure that the lane departure warning oscillating torque amplitude is below Max_Torque_Amplitude	x		
Functional_Safety_Requirement_02	...	...	...	...

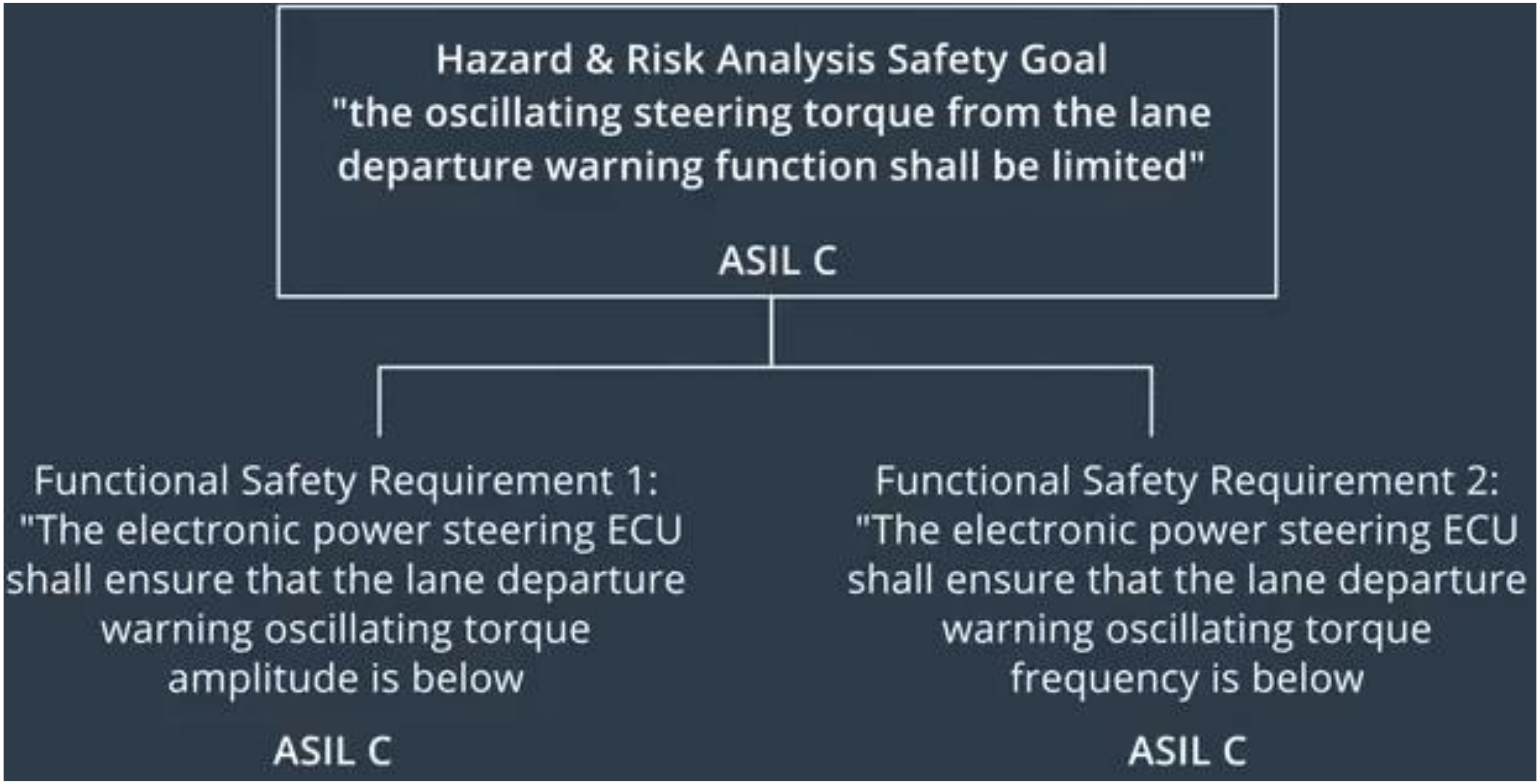
Architectural Refinement

- To implement the requirements, we add software blocks to the 3 ECUs (Electronic Power Steering, Camera Sensor, Car Display).
- Requirements may also be implemented with mechanical components instead of software blocks


Camera Sensor
Camera Sensor ECU
Lane Sensing
Torque request generator
Car Display ECU
Lane Assistance On/Off Status
Lane Assistance Active/ Inactive
Car Display
Driver Steering Torque Sensor
Electronic Power Steering ECU
Lane Assistance Functionality
Analyze Driver Steering Torque
Final Electronic Power Steering Torque Output
Steering Wheel
Motor providing torque to steering wheel

ASIL Inheritance

- The 2 functional safety requirements inherit ASIL C from the safety goal.



Hazard & Risk Analysis Safety Goal
"the oscillating steering torque from the lane departure warning function shall be limited"

ASIL C

Functional Safety Requirement 1:
"The electronic power steering ECU shall ensure that the lane departure warning oscillating torque amplitude is below

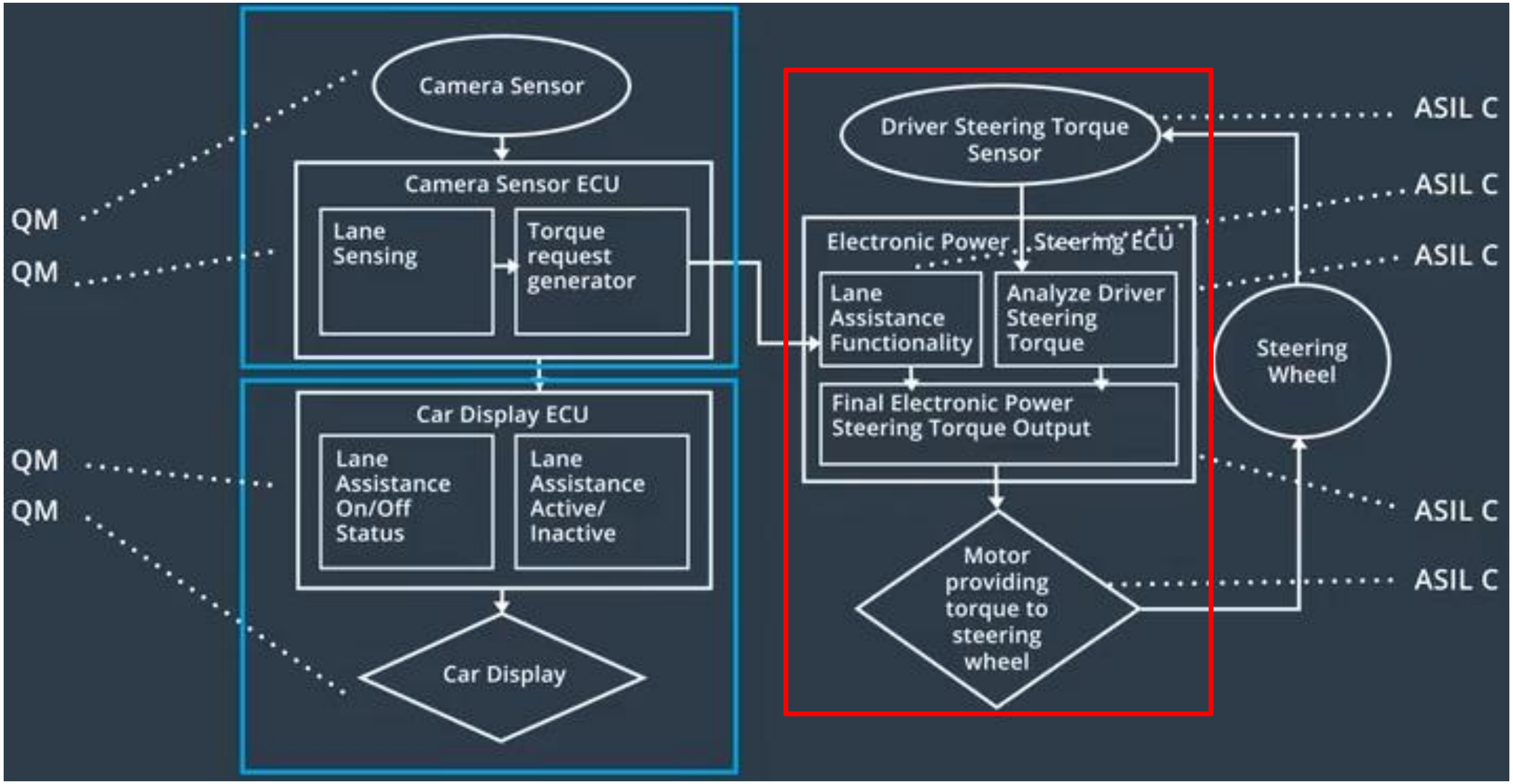
ASIL C

Functional Safety Requirement 2:
"The electronic power steering ECU shall ensure that the lane departure warning oscillating torque frequency is below

ASIL C

Allocation of ASIL

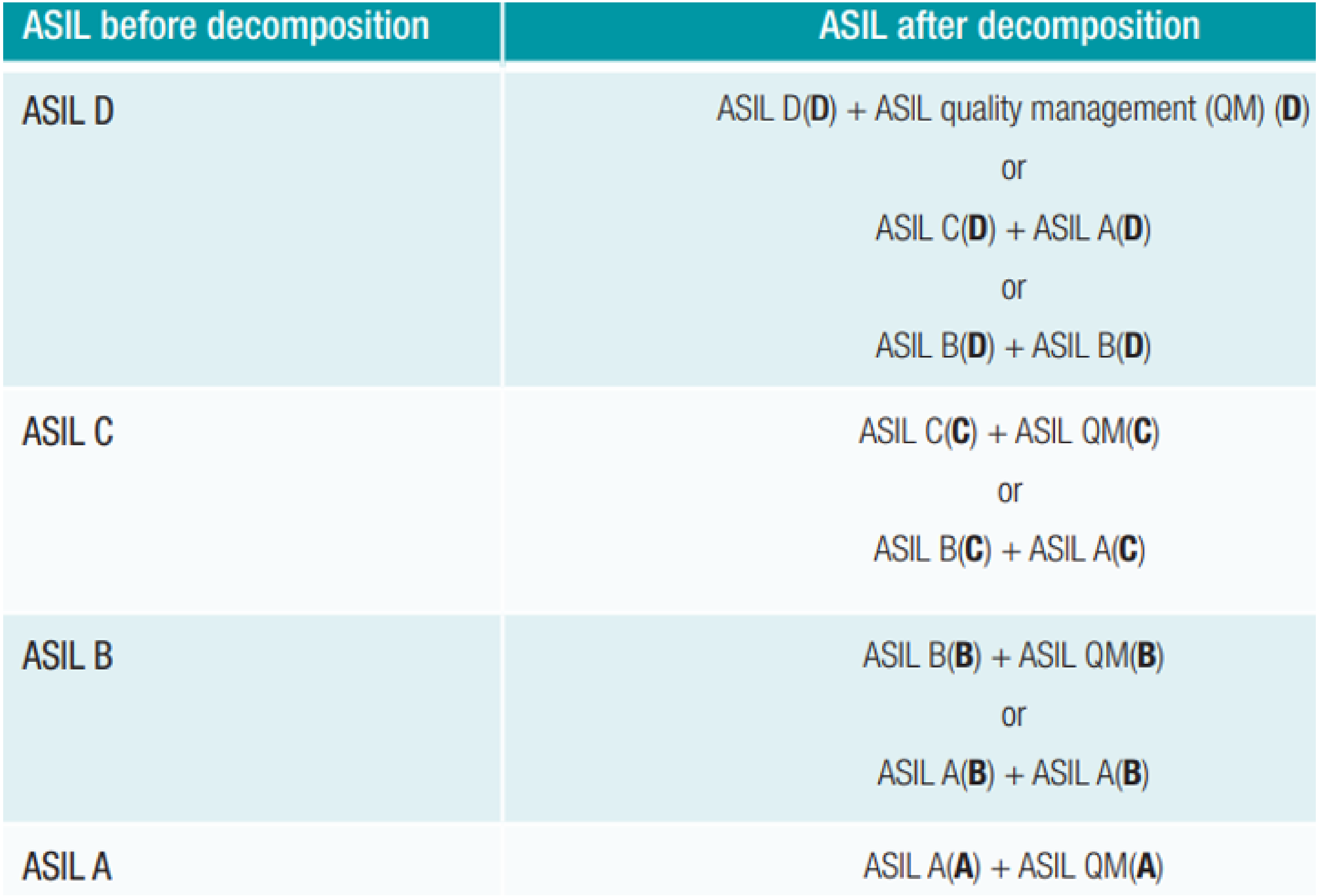
- The entire Power-Steering subsystem (ECU, sensor and motor) inherits ASIL-C from the 2 functional safety requirements.
- Since no functional safety requirements are allocated to the Camera Sensor and Car Display subsystems, they are not relevant to functional safety analysis, and are assigned QM (Quality Management).


Camera Sensor
Camera Sensor ECU
Lane Sensing
Torque request generator
Car Display ECU
Lane Assistance On/Off Status
Lane Assistance Active/ Inactive
Car Display
QM
QM
QM
QM
Driver Steering Torque Sensor
Electronic Power Steering ECU
Lane Assistance Functionality
Analyze Driver Steering Torque
Final Electronic Power Steering Torque Output
Motor providing torque to steering wheel
Steering Wheel
ASIL C
ASIL C
ASIL C
ASIL C
ASIL C

ASIL Decomposition

- ASIL decomposition can be used to lower ASIL, e.g., an ASIL D system can be decomposed into two redundant ASIL B subsystems.
  - If each subsystem has $P_1(malfunction) = P_2(malfunction) = .01$, then the overall system has $P(malfunction) = P_1(malfunction) * P_2(malfunction) = .0001$, assuming independent failure modes.
  - Also a motivation for sensor fusion.
- The subsystems are labeled as ASIL B(D).
  - In parenthesis is the parent ASIL before decomposition

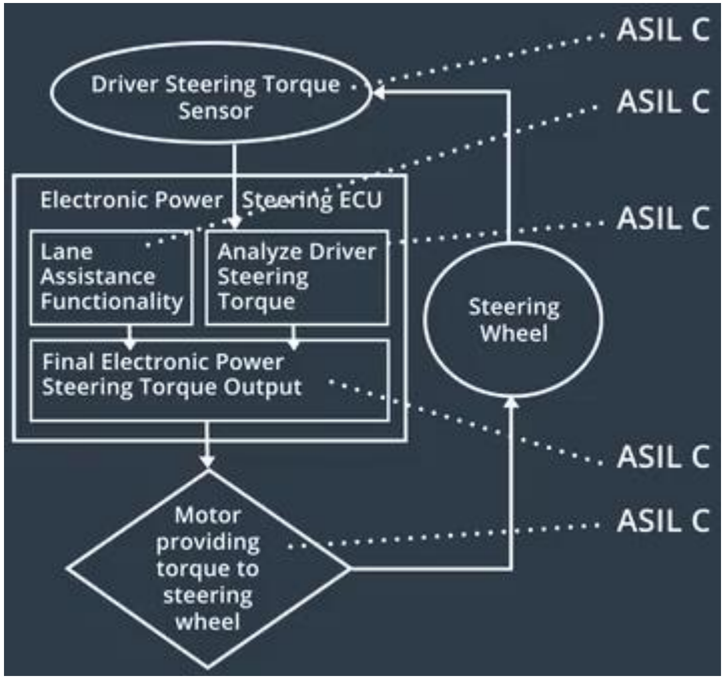
ISO 26262 ASIL Decomposition Rules

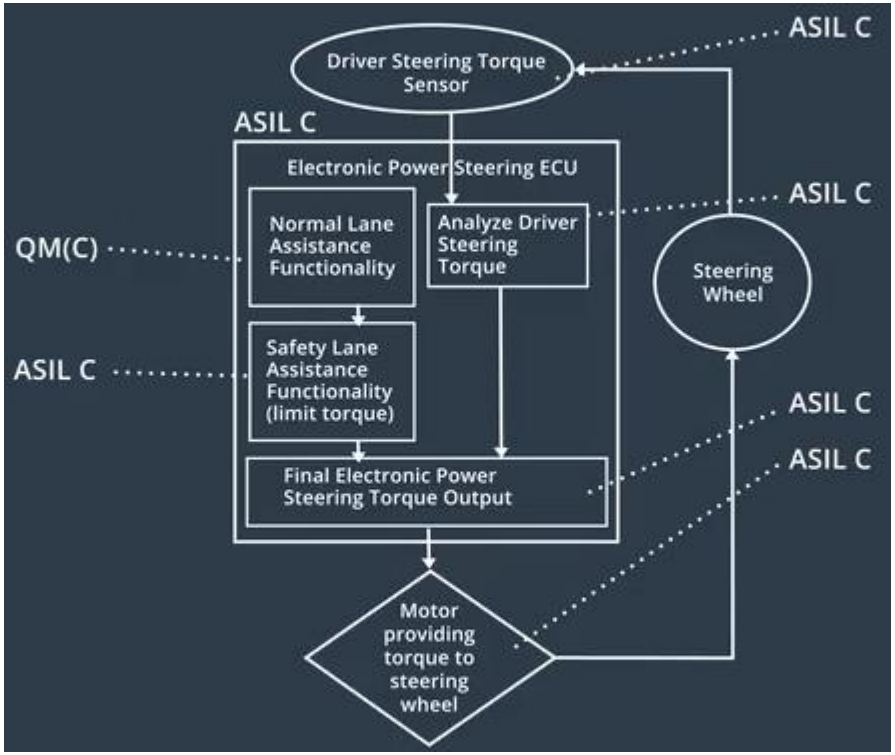


ASIL before decomposition	ASIL after decomposition
ASIL D	ASIL D(**D**) + ASIL quality management (QM) (**D**) or ASIL C(**D**) + ASIL A(**D**) or ASIL B(**D**) + ASIL B(**D**)
ASIL C	ASIL C(**C**) + ASIL QM(**C**) or ASIL B(**C**) + ASIL A(**C**)
ASIL B	ASIL B(**B**) + ASIL QM(**B**) or ASIL A(**B**) + ASIL A(**B**)
ASIL A	ASIL A(**A**) + ASIL QM(**A**)

ASIL Decomposition for Our Example

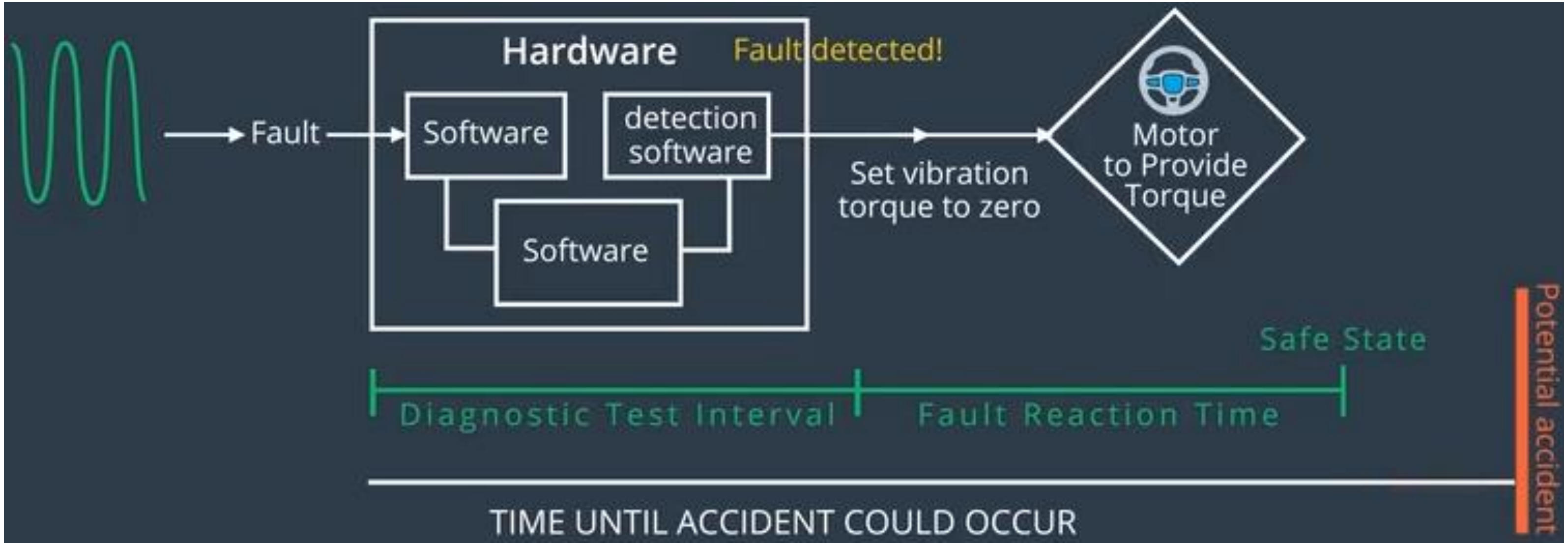
- We decompose the software block Lane Assistance Functionality (ASIL C) into 2 software blocks:
  - QM(C): Normal Lane Assistance Functionality
  - ASIL C(C): Safety Lane Assistance Functionality (limit torque)
- Since the Safety Lane Assistance Functionality is much smaller and simpler than the Normal Lane Assistance Functionality, this decomposition significantly reduces the burden of system design and analysis at ASIL C level.


ASIL C
Driver Steering Torque Sensor
ASIL C
Electronic Power Steering ECU
ASIL C
Lane Assistance Functionality
Analyze Driver Steering Torque
Steering Wheel
Final Electronic Power Steering Torque Output
ASIL C
Motor providing torque to steering wheel
ASIL C


ASIL C
Driver Steering Torque Sensor
ASIL C
Electronic Power Steering ECU
ASIL C
QM(C)
Normal Lane Assistance Functionality
Analyze Driver Steering Torque
Steering Wheel
ASIL C
Safety Lane Assistance Functionality (limit torque)
ASIL C
ASIL C
Final Electronic Power Steering Torque Output
Motor providing torque to steering wheel

Fault-Tolerance Time Interval (FTTI)

- FTTI for a functional safety requirement refers to the time it takes to detect and react to a malfunction.
- For example, for functional safety requirement 1:
  - The Electronic Power Steering ECU shall ensure that the lane departure oscillating torque amplitude is below Max_Torque_Amplitude.
- Fault: vibration (oscillating) torque amplitude exceeds the given threshold
- The safe state with acceptable risk level will be reached after FTTI, which is equal to sum of:
  - Diagnostic Test Interval: amount of time needed to detect the fault.
  - Fault Reaction Time: amount of time needed to react, e.g., by setting the vibration torque to zero.
- The FTTI must not exceed the time until accident could occur, say 50 ms.

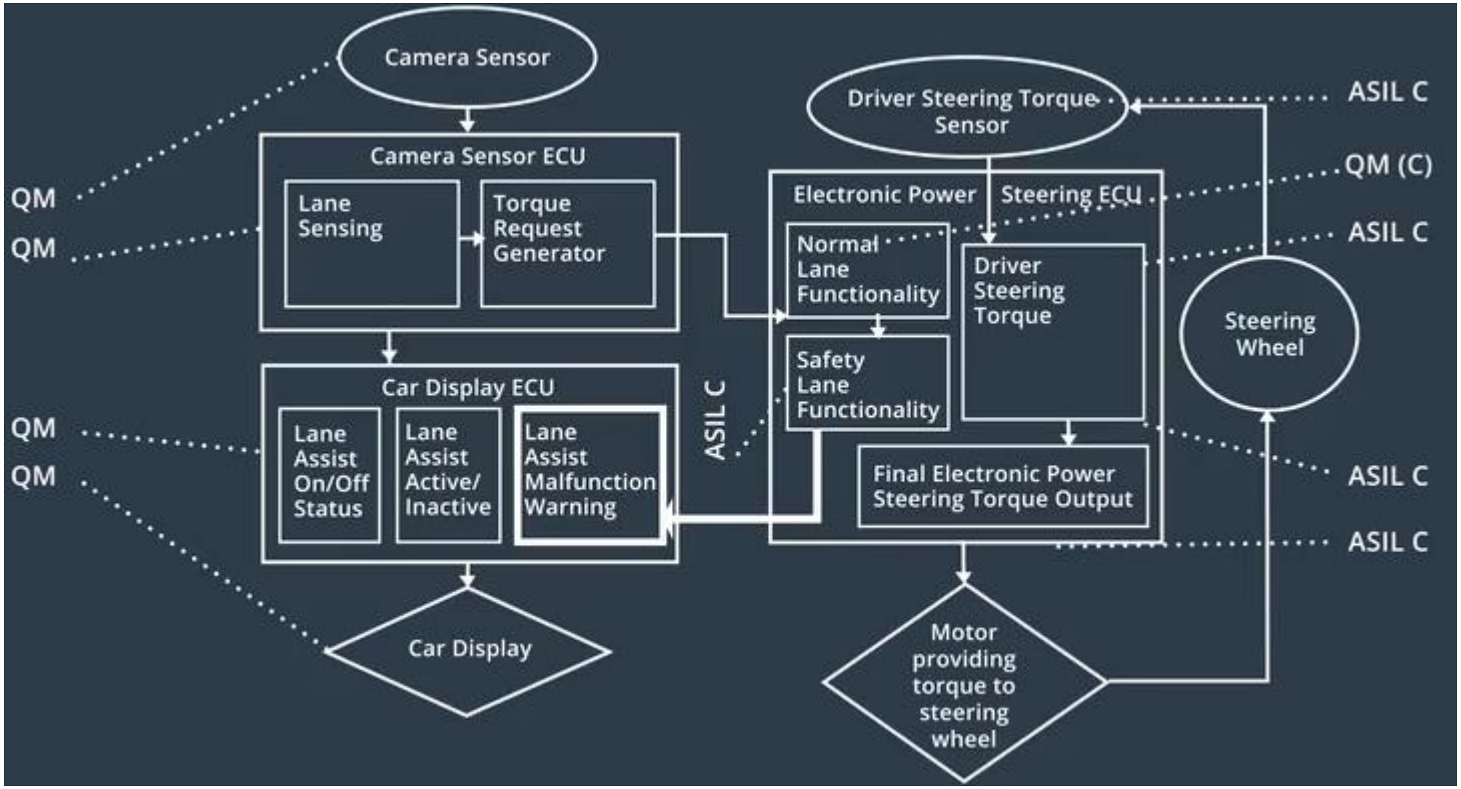


Warning and Degradation Concept

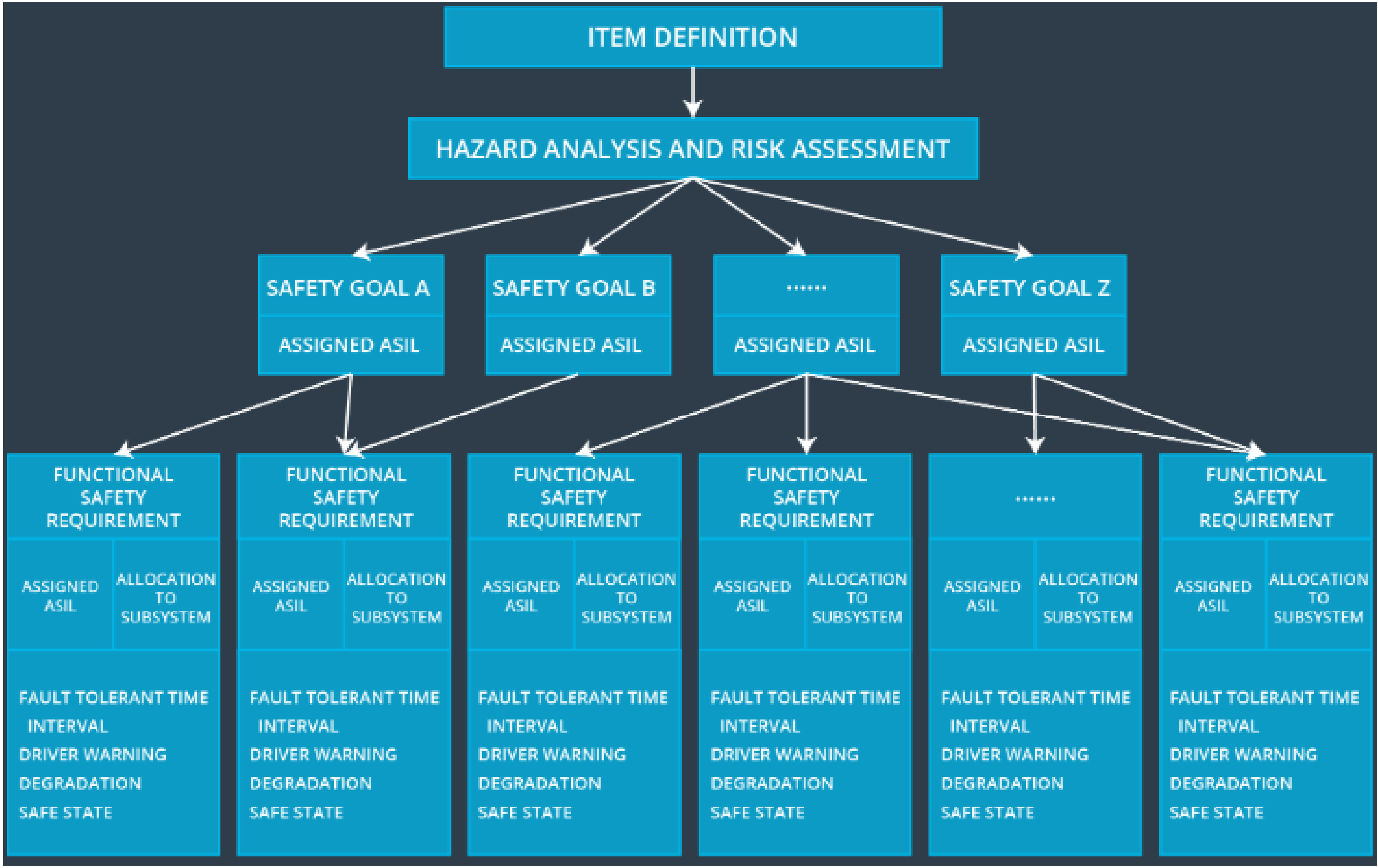
- It discusses:
  - how the driver will be warned of a malfunction.
  - what the system will do to "degrade" the functionality i.e. take the system to a safe state and recover from a safe state.
- Gradual Degradation vs. Turning a System Off
  - LDW and LKA are not critical for driving a vehicle. So if the system has a malfunction, we can turn them off to take the system to a safe state.
  - Turning off the system is not the only option. Some systems can provide limited functionality and "degrade" to different levels depending how bad the malfunction is. A car engine control system is one example. If one sensor fails, the engine control system might reduce the torque output of the motor so that the vehicle can still be driven but at a lower speed. Degrading the engine control system to a safer, but functioning, state would help the driver avoid getting stranded.

Driver Warning

- In case of malfunction of functional safety requirement (1 or 2), turn on the warning light to the driver.
- This is implemented by adding a software block Lane Assist Malfunction Warning, and a connection from Safety Lane Functionality to it.


Camera Sensor
Camera Sensor ECU
Lane Sensing
Torque Request Generator
Car Display ECU
Lane Assist On/Off Status
Lane Assist Active/ Inactive
Lane Assist Malfunction Warning
Car Display
QM
QM
QM
QM
ASIL C
Driver Steering Torque Sensor
Electronic Power Steering ECU
Normal Lane Functionality
Safety Lane Functionality
Driver Steering Torque
Final Electronic Power Steering Torque Output
Steering Wheel
Motor providing torque to steering wheel
ASIL C
QM (C)
ASIL C
ASIL C
ASIL C

Functional Safety Analysis Workflow

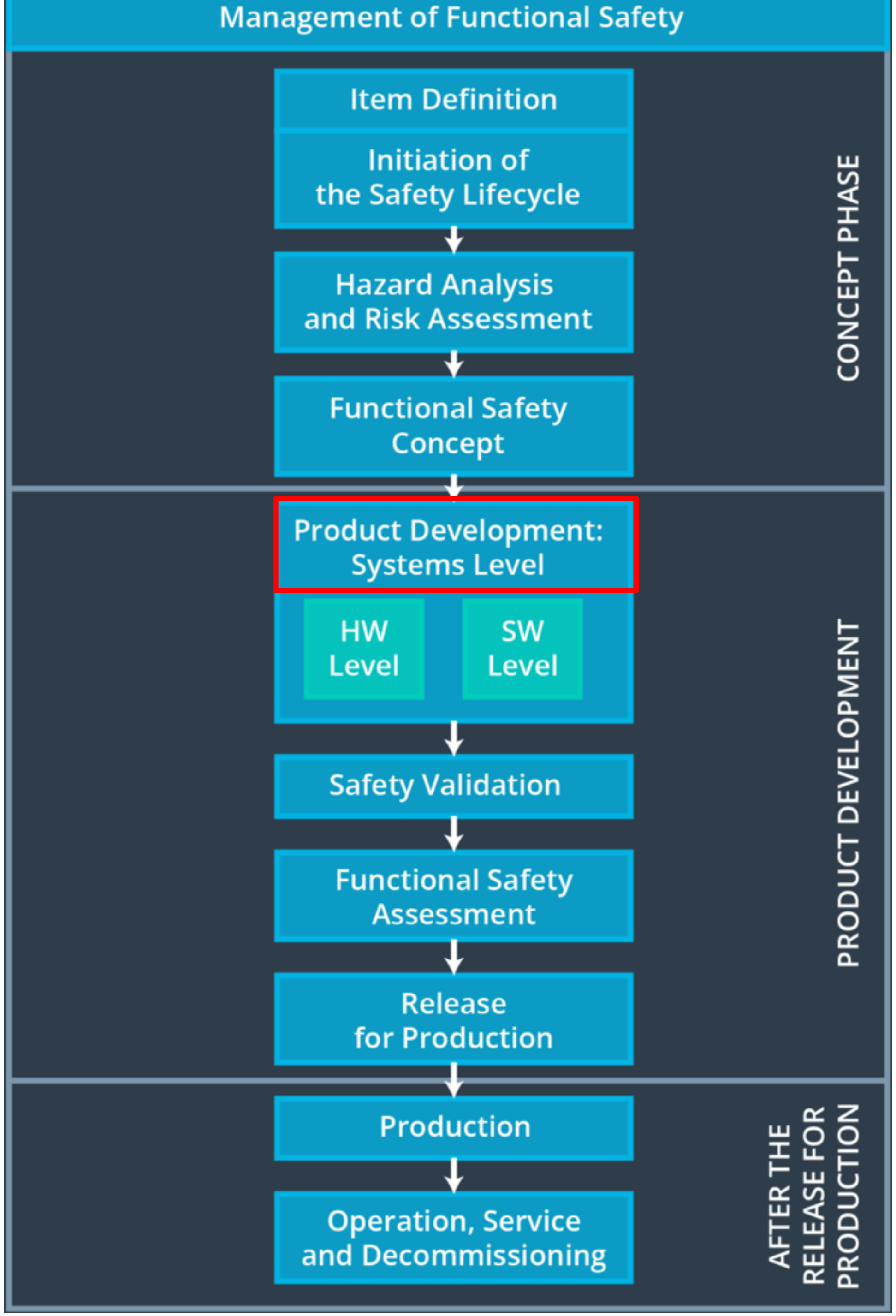


Outline

- Introduction to functional safety
- Hazard Analysis and Risk Assessment (HARA)
- Functional Safety Concept
- Technical Safety Concept
- Functional Safety for Hardware and Software

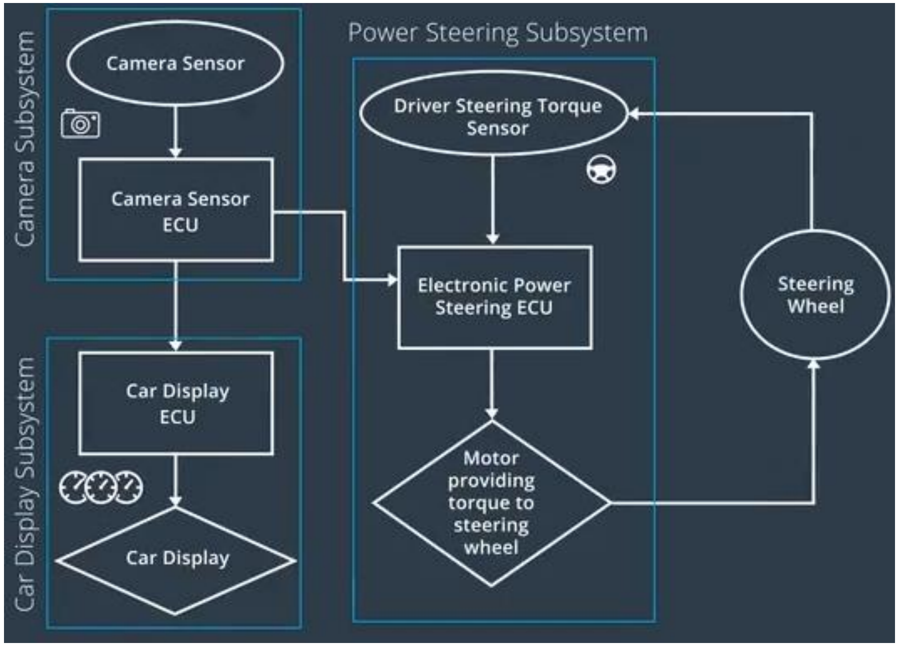
Technical Safety Concept

- The functional safety concept is the last step in the concept phase
- The product development phase is divided into two parts:
  - Product development at the system level (here lies technical safety concept)
  - Product development at the hardware and software level


Management of Functional Safety
Item Definition
Initiation of the Safety Lifecycle
Hazard Analysis and Risk Assessment
Functional Safety Concept
CONCEPT PHASE
Product Development: Systems Level
HW Level
SW Level
Safety Validation
Functional Safety Assessment
Release for Production
PRODUCT DEVELOPMENT
Production
Operation, Service and Decommissioning
AFTER THE RELEASE FOR PRODUCTION

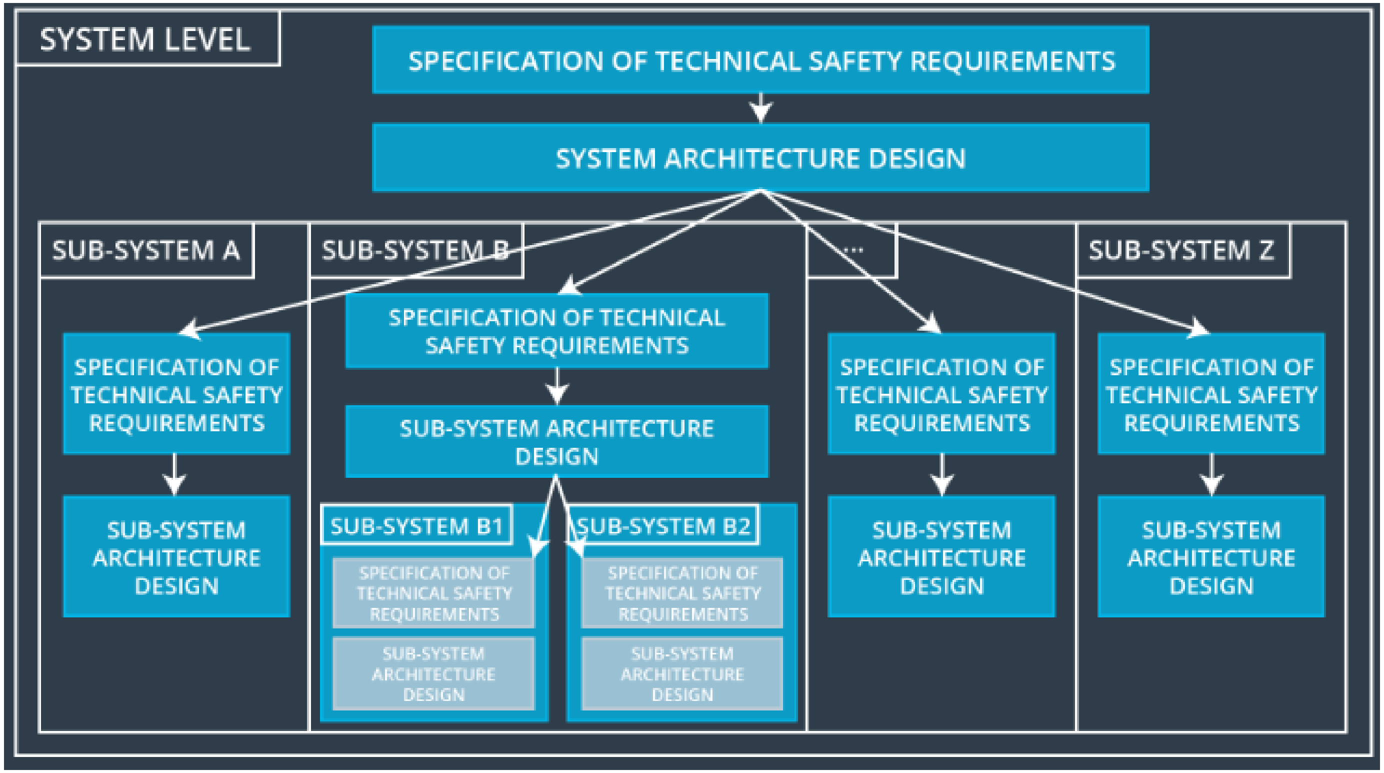
Technical Safety Concept

- The technical safety concept involves 2 steps:
  - Turning functional safety requirements into technical safety requirements for each of these subsystems (Camera, Display, Power Steering).
  - Allocating technical safety requirements to the system architecture.



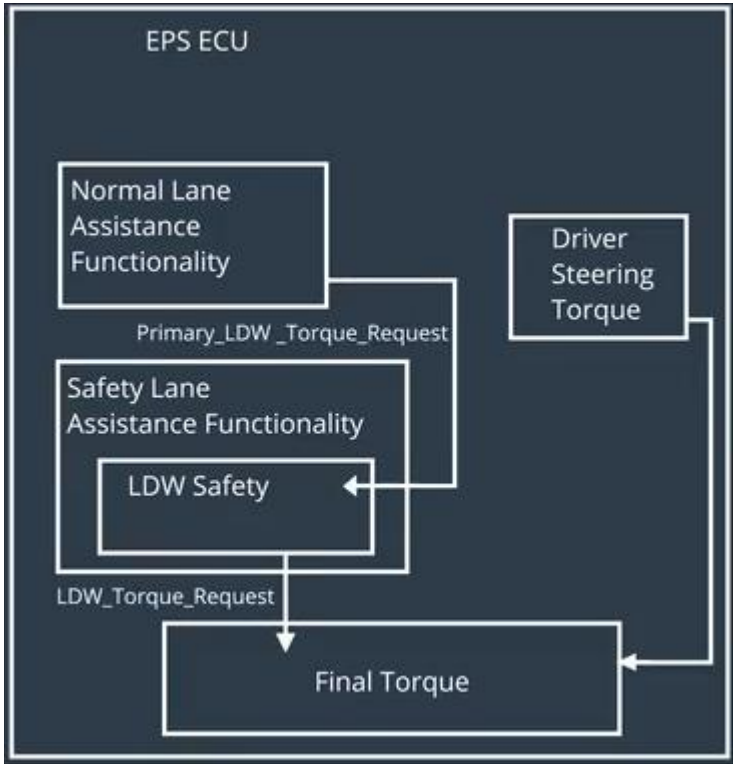
Technical Safety Concept at the System Level vs. ECU Level

- In our simple example, the only safety relevant element was the Electronic Power Steering ECU; thus, our technical safety concept will go directly to the ECU level. But in general, you will need to develop multiple technical safety concepts; one for each safety-relevant subsystem, and then one for each safety-relevant ECU.
- Figure shows how a technical safety concept might be divided into several documents for different subsystems.


SYSTEM LEVEL
SPECIFICATION OF TECHNICAL SAFETY REQUIREMENTS
SYSTEM ARCHITECTURE DESIGN
SUB-SYSTEM A
SUB-SYSTEM B
...
SUB-SYSTEM Z
SPECIFICATION OF TECHNICAL SAFETY REQUIREMENTS
SUB-SYSTEM ARCHITECTURE DESIGN
SPECIFICATION OF TECHNICAL SAFETY REQUIREMENTS
SUB-SYSTEM ARCHITECTURE DESIGN
SUB-SYSTEM B1
SUB-SYSTEM B2
SPECIFICATION OF TECHNICAL SAFETY REQUIREMENTS
SUB-SYSTEM ARCHITECTURE DESIGN
SPECIFICATION OF TECHNICAL SAFETY REQUIREMENTS
SUB-SYSTEM ARCHITECTURE DESIGN
SPECIFICATION OF TECHNICAL SAFETY REQUIREMENTS
SUB-SYSTEM ARCHITECTURE DESIGN
SPECIFICATION OF TECHNICAL SAFETY REQUIREMENTS
SUB-SYSTEM ARCHITECTURE DESIGN

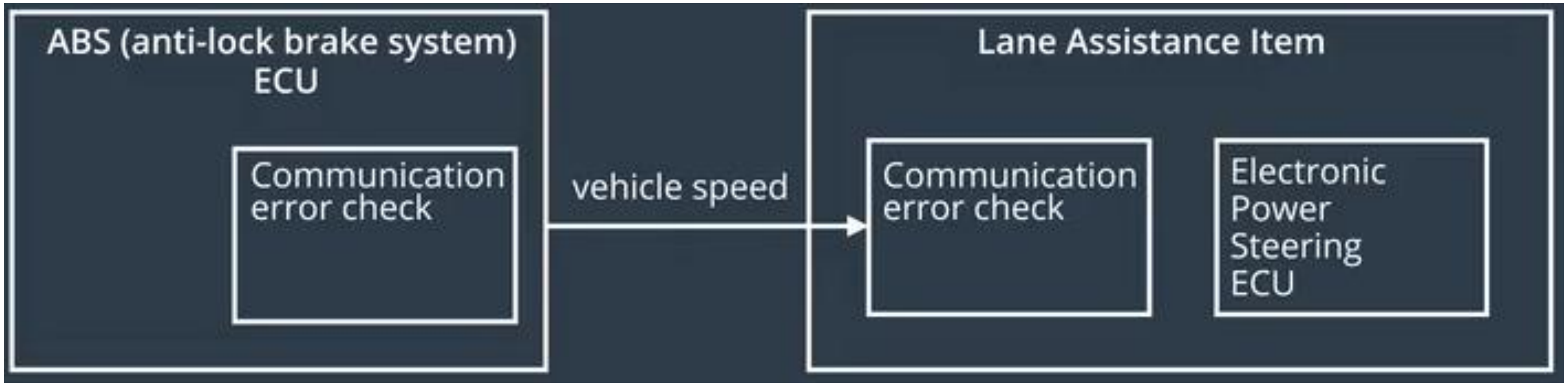
Functional vs. Technical Safety Requirements

- Functional safety requirement is abstract and high-level:
  - “The lane keeping item shall ensure that the lane departure oscillating torque amplitude is below Max_Torque_Amplitude.”
- Technical safety requirement is more concrete, and describes the signal flow, and which component is in charge of the functionality:
  - “The LDW safety component shall ensure that the amplitude of the ‘LDW_Torque_Request’ sent to the ‘Final torque’ component is below ‘Max_Torque_Amplitude’.”
  - This can be implemented by setting ‘LDW_Torque_Request’ to 0 if it exceeds ‘Max_Torque_Amplitude’, but we are not concerned with implementation yet.


EPS ECU
Normal Lane Assistance Functionality
Driver Steering Torque
Primary_LDW _Torque_Request
Safety Lane Assistance Functionality
LDW Safety
LDW_Torque_Request
Final Torque

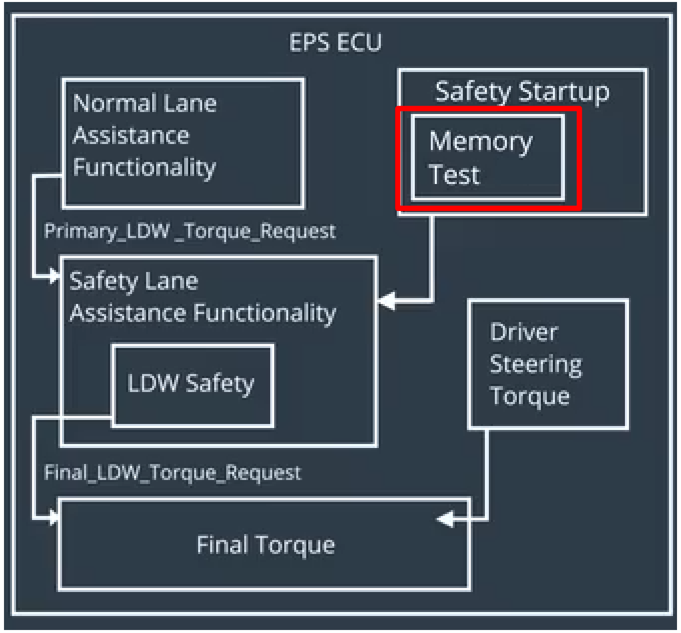
Additional Categories of Technical Safety Requirements

- 1. Detecting faults within a system
  - “The validity and integrity of the data transmission for ‘LDW_Torque_Request’ signal shall be ensured.”
- 2. Detecting faults in an external device interacting with the system
  - The Anti-Lock Brake ECU might send vehicle speed data to the Electronic Power Steering ECU. CRCs (Cyclic Redundancy checks) or alive counters can be used to detect fault in data transmission.


ABS (anti-lock brake system) ECU
Communication error check
vehicle speed
Lane Assistance Item
Communication error check
Electronic Power Steering ECU

Additional Categories of Technical Safety Requirements

- 3. Reaching a safe state
  - “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.”
- 4. Implementing a warning and degradation concept
  - “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.”
- 5. Preventing latent faults
  - “Memory test shall be conducted at start up of the EPS ECU to check for any faults in memory.”



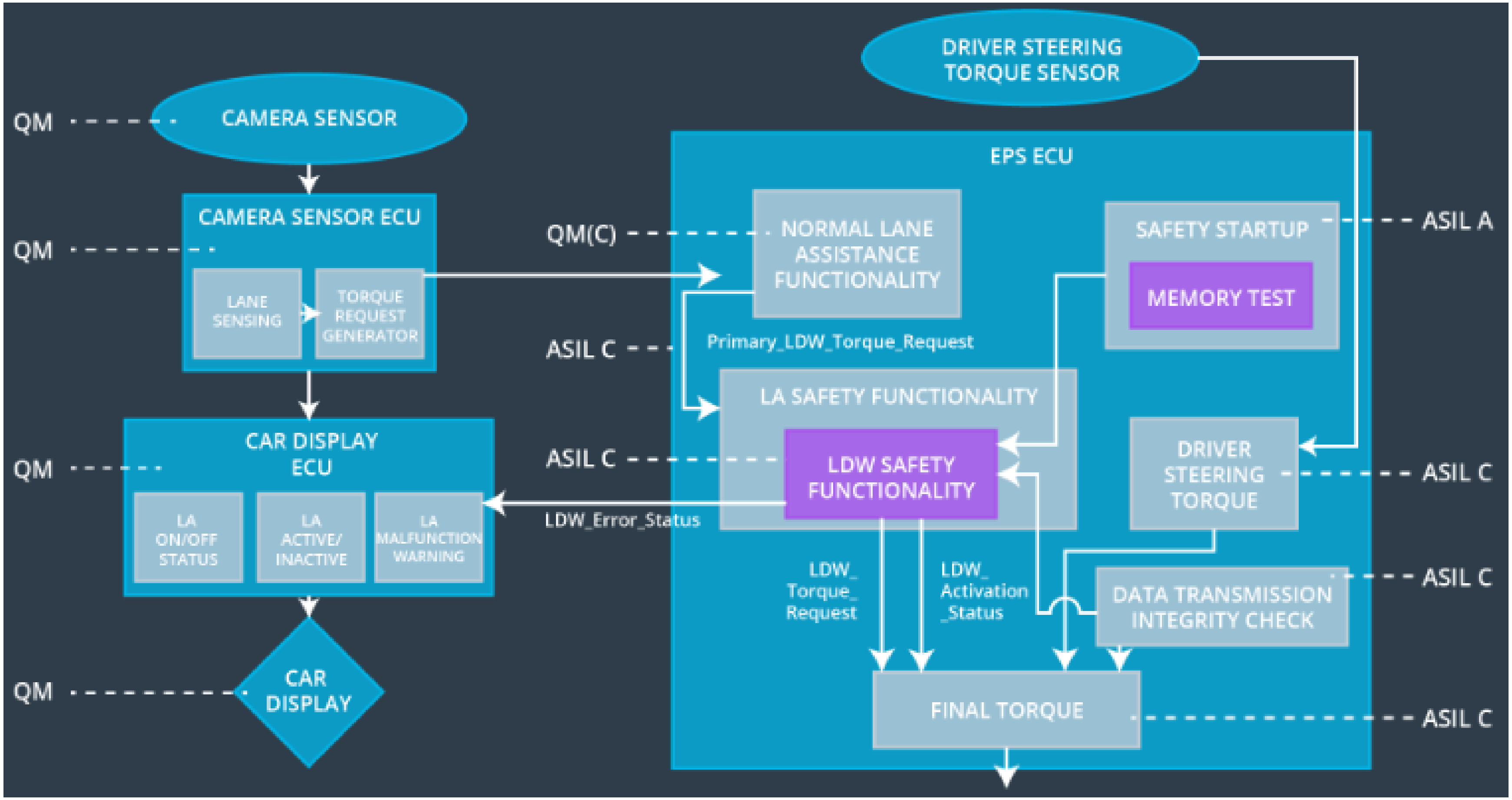
Technical Safety Requirement Examples

- Technical safety requirement inherits ASIL from its corresponding functional safety requirement.
  - TechSafReq01 to 04 are all based on the functional safety requirement “The lane keeping item shall ensure that the lane departure oscillating torque amplitude is below Max_Torque_Amplitude.” with ASIL C, hence they are all ASIL C.
  - TechSafReq05 is ASIL A, since ISO 26262 says that “latent fault detection for ASIL C requirement can be labeled ASIL A.”

Req ID	Technical Safety Requirement.	FTTI	ASIL
TechSafReq01	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.	50 ms	C
TechSafReq02	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.	50 ms	C
TechSafReq03	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.	50 ms	C
TechSafReq04	The validity and integrity of the data transmission for 'LDW_Torque_Request' signal shall be ensured.	50 ms	C
TechSafReq05	Memory test shall be conducted at start up of the EPS ECU to check for any faults in memory.	ignition cycle	A

Allocation of Requirements to System Architecture

- TechSafReq01 to 05 have been allocated to different software elements such as the "LDW Safety Functionality" block, the "Data Transmission Integrity Check", or other relevant blocks inside the EPS ECU.



Allocation of Requirements to System Architecture: Notable Points

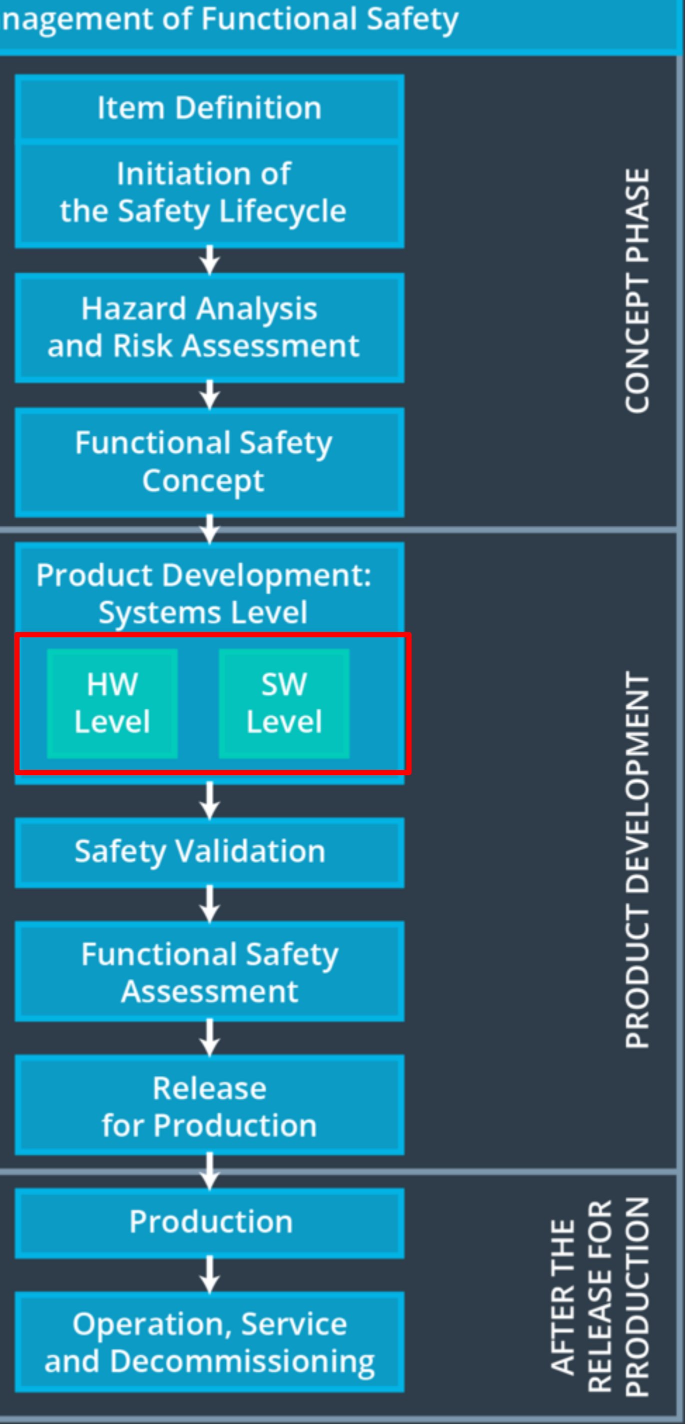
- Elements inherit the highest ASIL from the technical safety requirements. So if the same software element provides functionality to the LDW function (ASIL C) and LKA function (ASIL B), the higher ASIL wins (ASIL C).
- If an element contains subelements with different ASILs, both sub-elements receive the highest ASIL. The exception is if the criteria for coexistence is met:
  - If a failure in one sub-element will not affect the other sub-element, then the sub-elements can have different ASILs.
- Internal and external interfaces for safety-related elements need to be clearly defined. This way non-safety related elements are clearly identified as well.

Outline

- Introduction to functional safety
- Hazard Analysis and Risk Assessment (HARA)
- Functional Safety Concept
- Technical Safety Concept
- Functional Safety for Hardware and Software

Functional Safety for Hardware and Software

- After so much requirements engineering, finally we are ready to develop some hardware and software!



Management of Functional Safety

Item Definition

Initiation of the Safety Lifecycle

Hazard Analysis and Risk Assessment

Functional Safety Concept

CONCEPT PHASE

Product Development: Systems Level

HW Level

SW Level

Safety Validation

Functional Safety Assessment

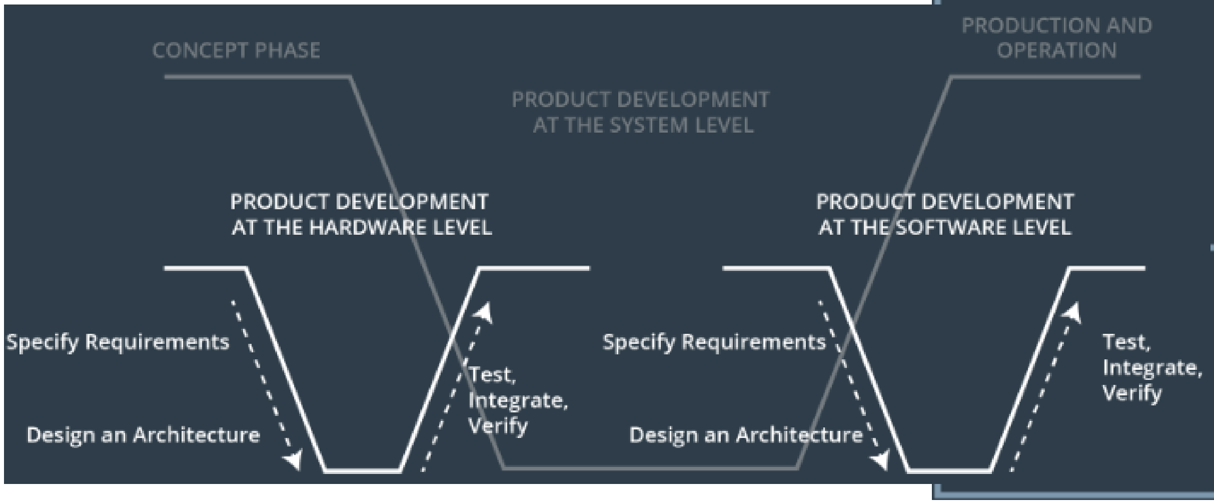
Release for Production

PRODUCT DEVELOPMENT

Production

Operation, Service and Decommissioning

AFTER THE RELEASE FOR PRODUCTION



CONCEPT PHASE

PRODUCT DEVELOPMENT AT THE SYSTEM LEVEL

PRODUCTION AND OPERATION

PRODUCT DEVELOPMENT AT THE HARDWARE LEVEL

PRODUCT DEVELOPMENT AT THE SOFTWARE LEVEL

Specify Requirements

Design an Architecture

Test, Integrate, Verify

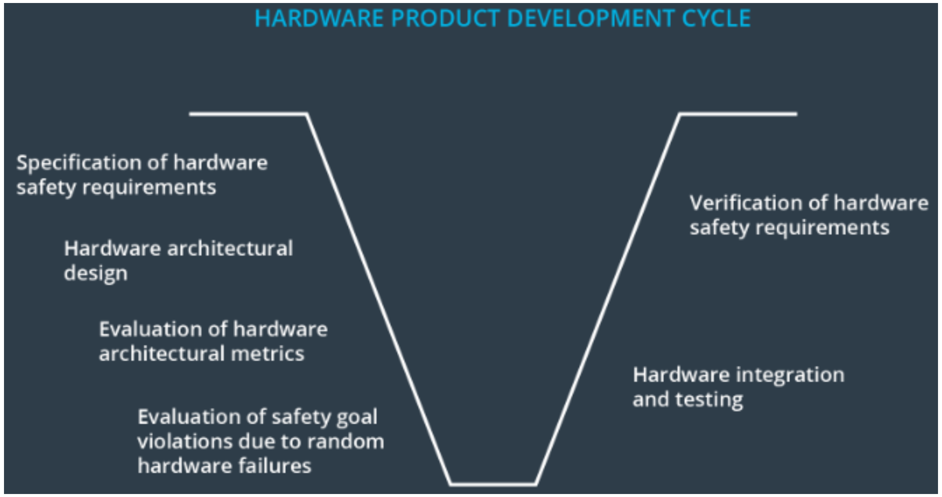
Specify Requirements

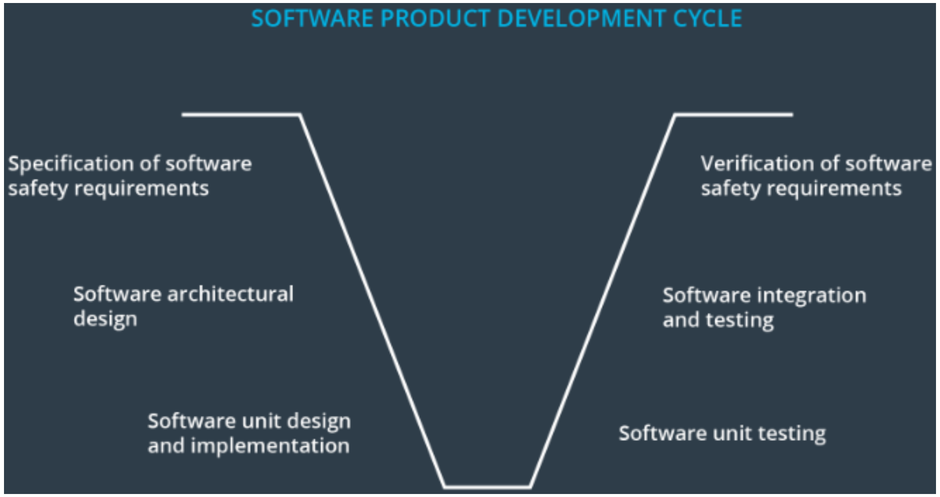
Design an Architecture

Test, Integrate, Verify

Hardware and Software Product Development Life-Cycle

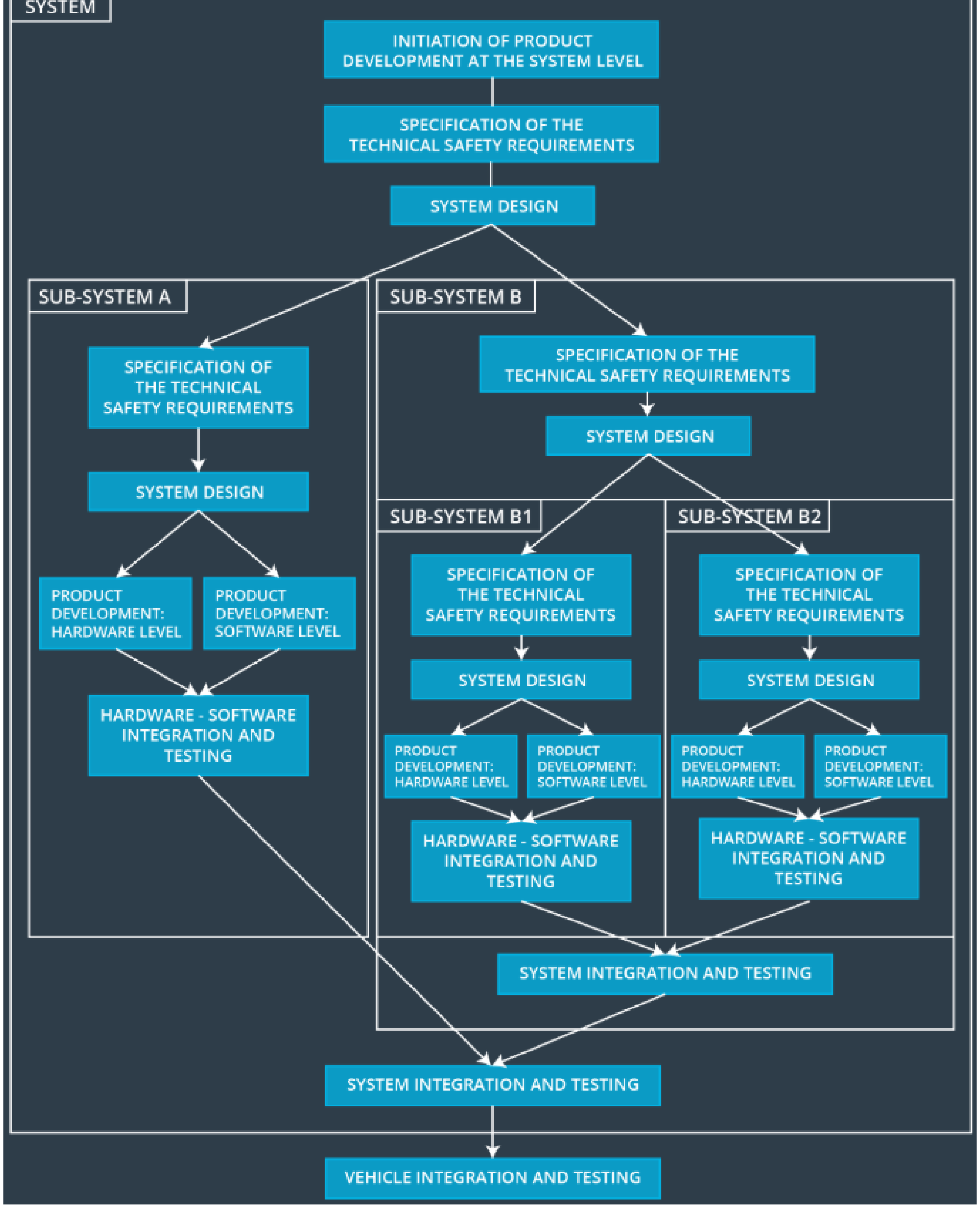
- The general idea represented in each V stays the same; first, you specify safety requirements. Then you allocate these requirements to a system architecture. Finally you test, integrate, and verify.
- But there is an extra step on the hardware and software sides.
  - For hardware, the V model includes sections about hardware architectural metrics and evaluation of random hardware failures.
  - For software, there is an architectural design section as well as a unit design section.
    - A unit is a smaller part of a software architecture, e.g., could be a software driver to read raw data from a camera sensor.


HARDWARE PRODUCT DEVELOPMENT CYCLE
Specification of hardware safety requirements
Hardware architectural design
Evaluation of hardware architectural metrics
Evaluation of safety goal violations due to random hardware failures
Verification of hardware safety requirements
Hardware integration and testing


SOFTWARE PRODUCT DEVELOPMENT CYCLE
Specification of software safety requirements
Software architectural design
Software unit design and implementation
Verification of software safety requirements
Software integration and testing
Software unit testing

Bird's Eye View

- Figure shows a general outline of what would be involved in a functional safety project. You can see that the steps of the V model have been stretched out vertically. Oftentimes an item will have multiple systems and subsystems. Subsystems will have their own software and hardware requirements. And these subsystems need to be integrated into larger systems:

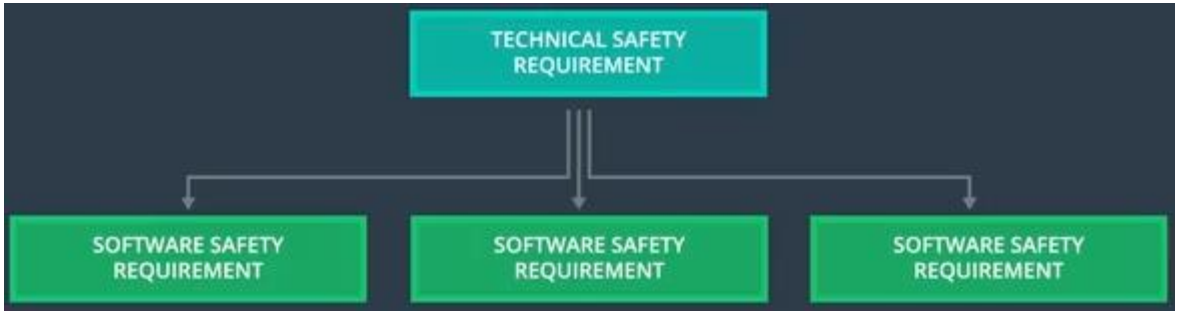

SYSTEM
INITIATION OF PRODUCT DEVELOPMENT AT THE SYSTEM LEVEL
SPECIFICATION OF THE TECHNICAL SAFETY REQUIREMENTS
SYSTEM DESIGN
SUB-SYSTEM A
SPECIFICATION OF THE TECHNICAL SAFETY REQUIREMENTS
SYSTEM DESIGN
PRODUCT DEVELOPMENT: HARDWARE LEVEL
PRODUCT DEVELOPMENT: SOFTWARE LEVEL
HARDWARE - SOFTWARE INTEGRATION AND TESTING
SUB-SYSTEM B
SPECIFICATION OF THE TECHNICAL SAFETY REQUIREMENTS
SYSTEM DESIGN
SUB-SYSTEM B1
SPECIFICATION OF THE TECHNICAL SAFETY REQUIREMENTS
SYSTEM DESIGN
PRODUCT DEVELOPMENT: HARDWARE LEVEL
PRODUCT DEVELOPMENT: SOFTWARE LEVEL
HARDWARE - SOFTWARE INTEGRATION AND TESTING
SUB-SYSTEM B2
SPECIFICATION OF THE TECHNICAL SAFETY REQUIREMENTS
SYSTEM DESIGN
PRODUCT DEVELOPMENT: HARDWARE LEVEL
PRODUCT DEVELOPMENT: SOFTWARE LEVEL
HARDWARE - SOFTWARE INTEGRATION AND TESTING
SYSTEM INTEGRATION AND TESTING
SYSTEM INTEGRATION AND TESTING
VEHICLE INTEGRATION AND TESTING

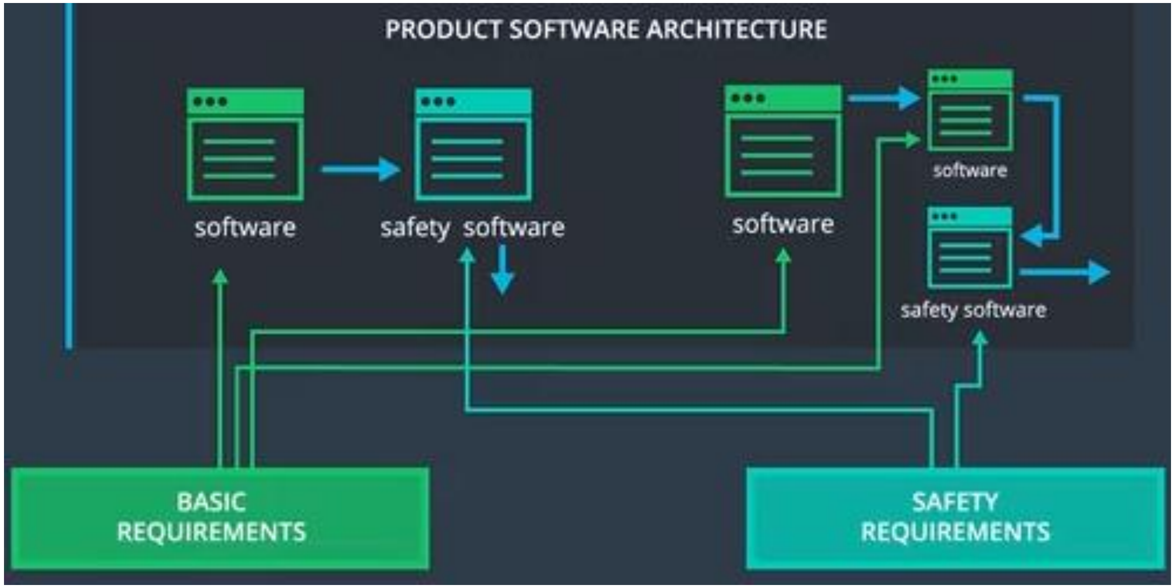
Programming Languages

- Popular languages: C, C++, Matlab (with automatic code generation to C/C++)
- Software development guidelines
  - – MISRA C, MISRA C++ specify a subset of C or C++ for safety-critical applications, including
    - Defensive implementation techniques
    - Language subsets
    - Style guides
    - Naming conventions
  - – (MISRA stands for Motor Industry Software Reliability Association)

Software Safety Requirements

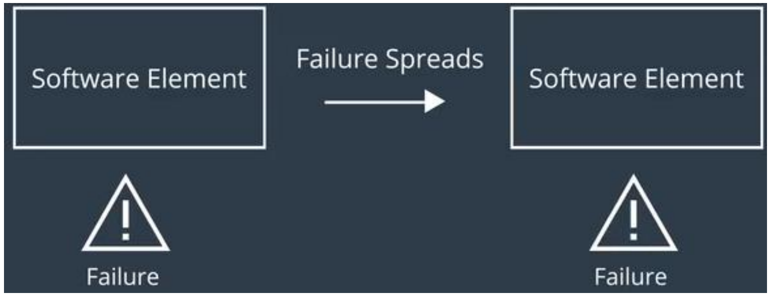
- Software Safety Requirements are derived from Technical Safety Requirements, and allocated to software architecture elements.

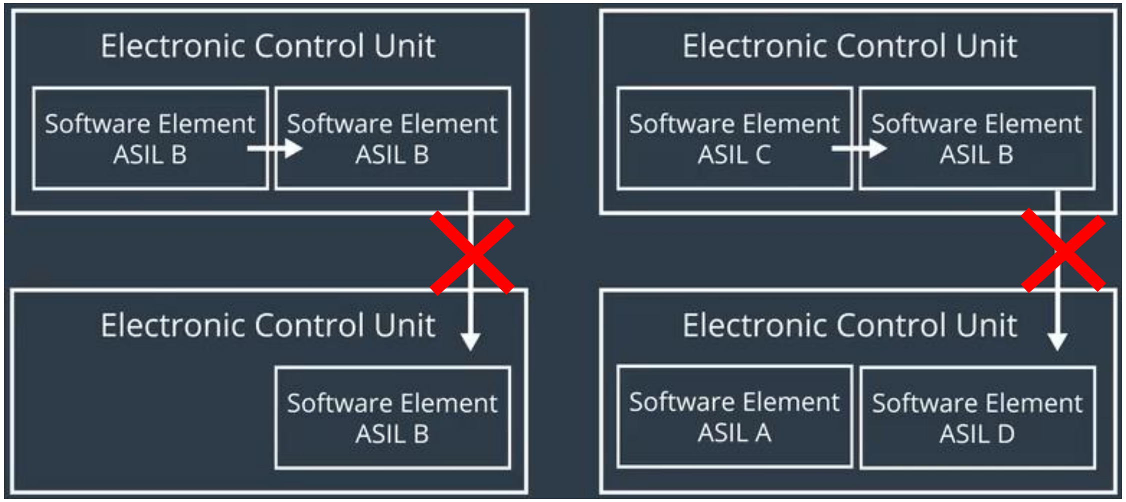

TECHNICAL SAFETY REQUIREMENT
SOFTWARE SAFETY REQUIREMENT
SOFTWARE SAFETY REQUIREMENT
SOFTWARE SAFETY REQUIREMENT



Freedom from Interference

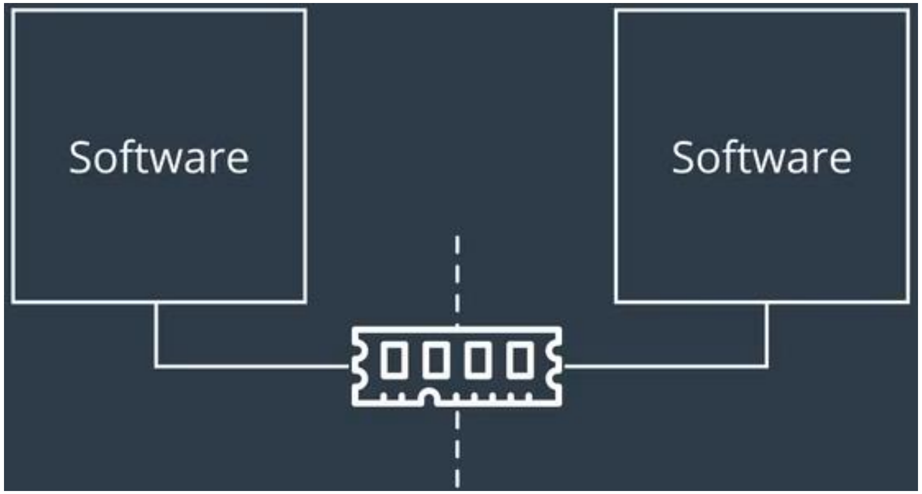
- Freedom from interference refers to the requirement that failure in one software element does not spread and cause failure in other software elements that communicate with it, or run on the same ECU.
- May be between software elements with the same ASIL, or different ASILs
- Interferences include
  - Spatial
  - Temporal

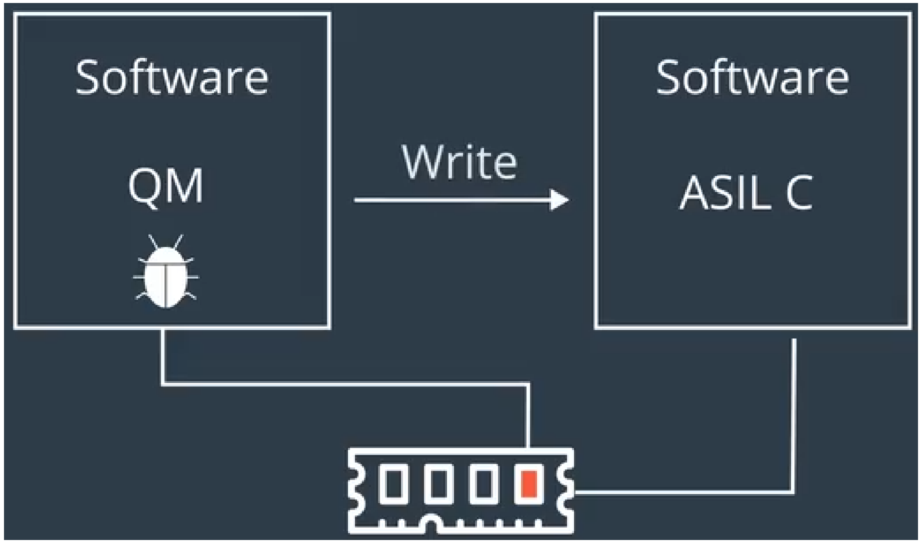




Spatial Interference

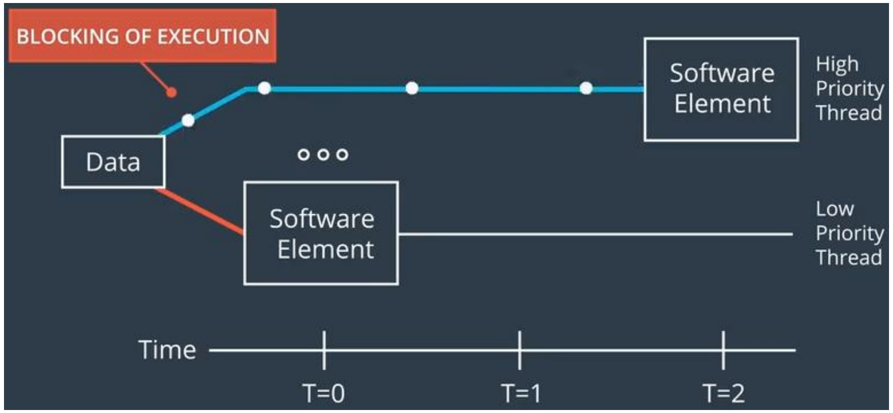
- Memory address spaces of software elements should be partitioned.
- A counter-example: a software element (QM) has a bug that creates a pointer to the memory address space of a software element (ASIL C), and writes to it, leading to safety goal violation.

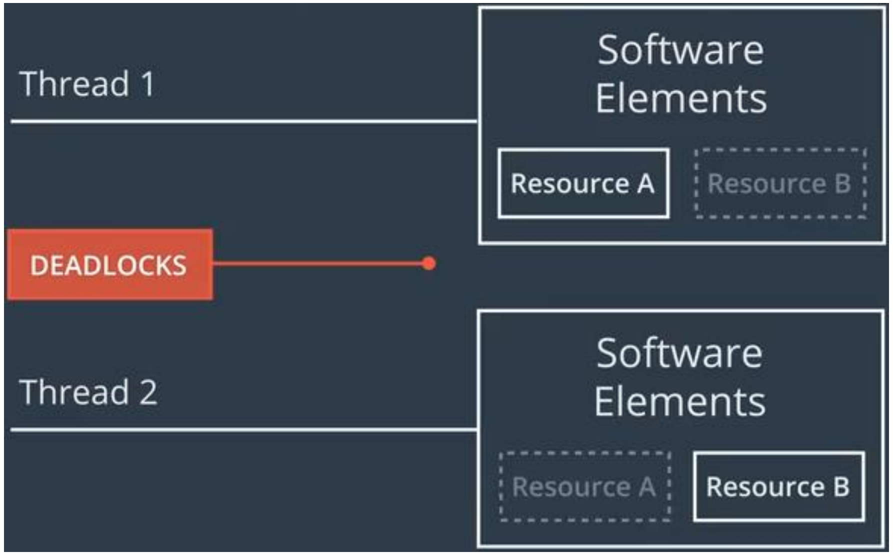


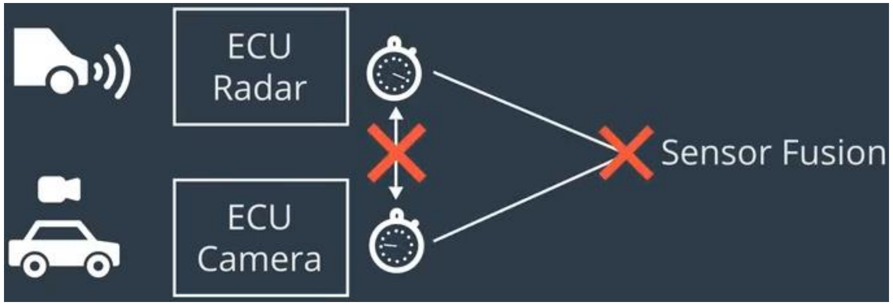


Temporal Interference

- Top: Blocking of execution due to shared data
- Middle: deadlock due to circular waiting:
  - Software element A is holding Resource A and requesting Resource B; Software element B is holding Resource B and requesting Resource A
- Bottom: incorrect sensor fusion due to incorrect clock synchronization between software elements

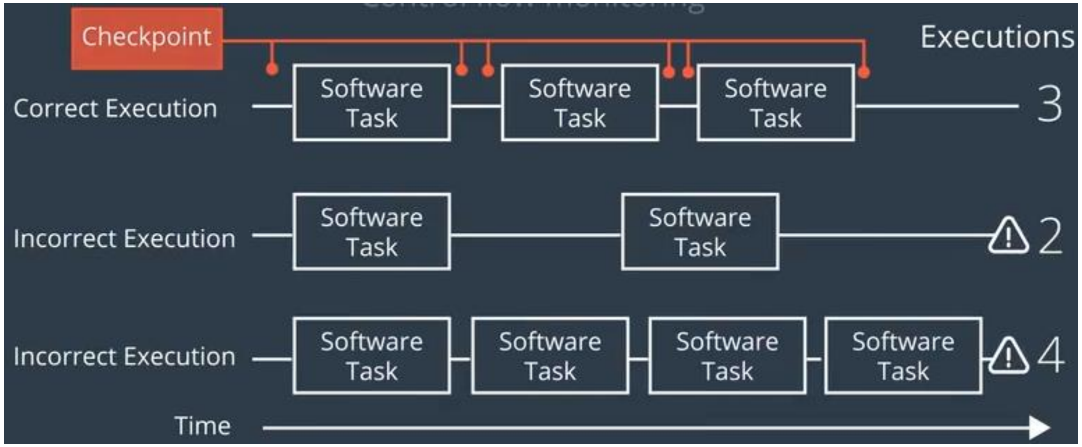


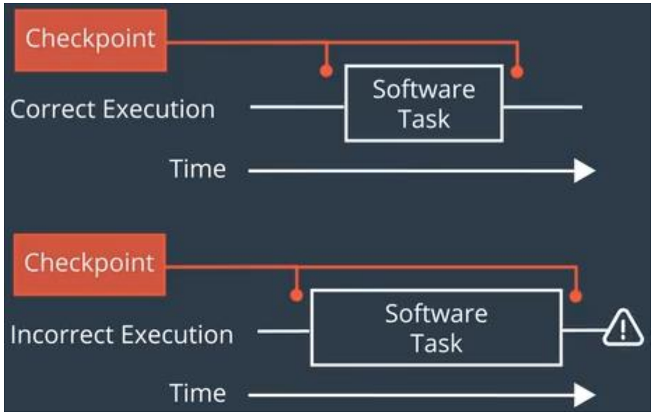


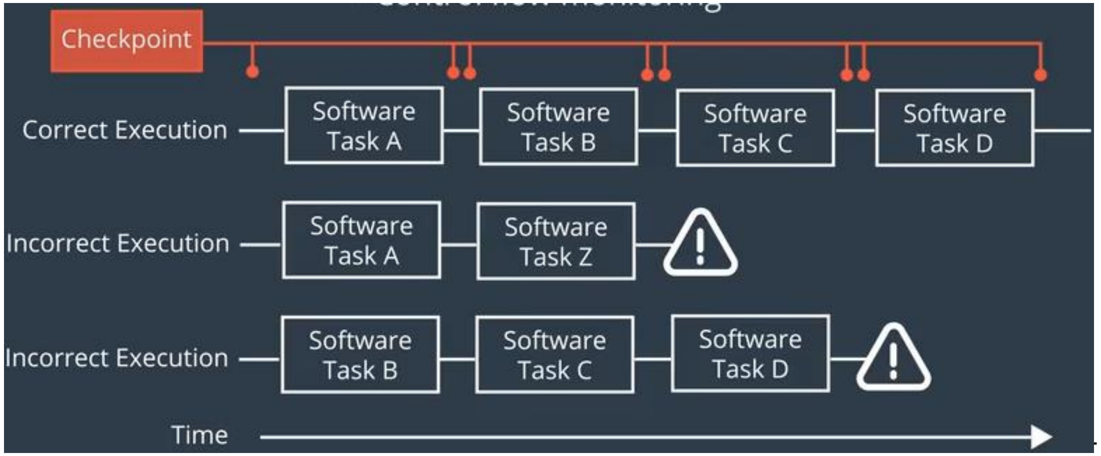


More Temporal Interference

- Incorrect allocation of execution time & incorrect execution of sequence
- Top: Alive supervision checks # times a software element is executed in a time span. (In the fig, 3 is correct, 2 or 4 is not)
- Middle: Deadline monitoring checks if a software element is executed before its deadline
- Bottom: Control Flow Monitoring checks that software elements are executed in the correct sequence

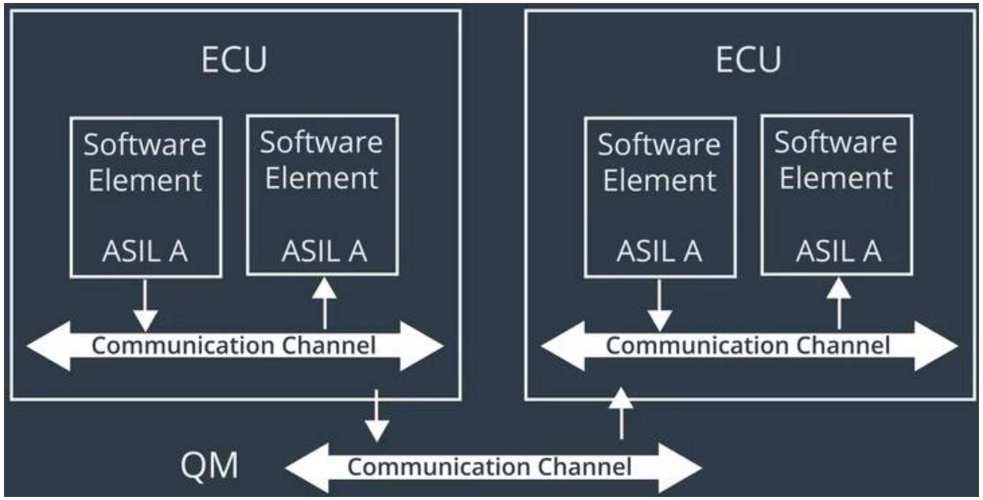


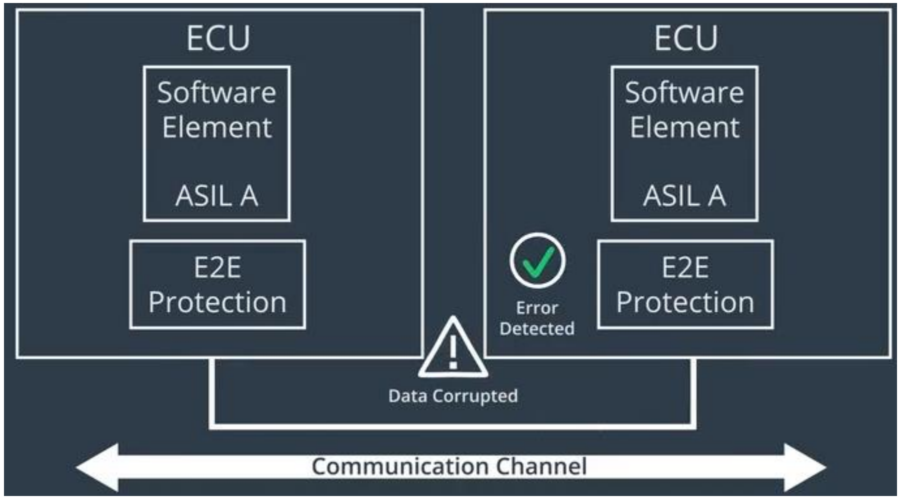




Communication Interference and End-to-End Protocol

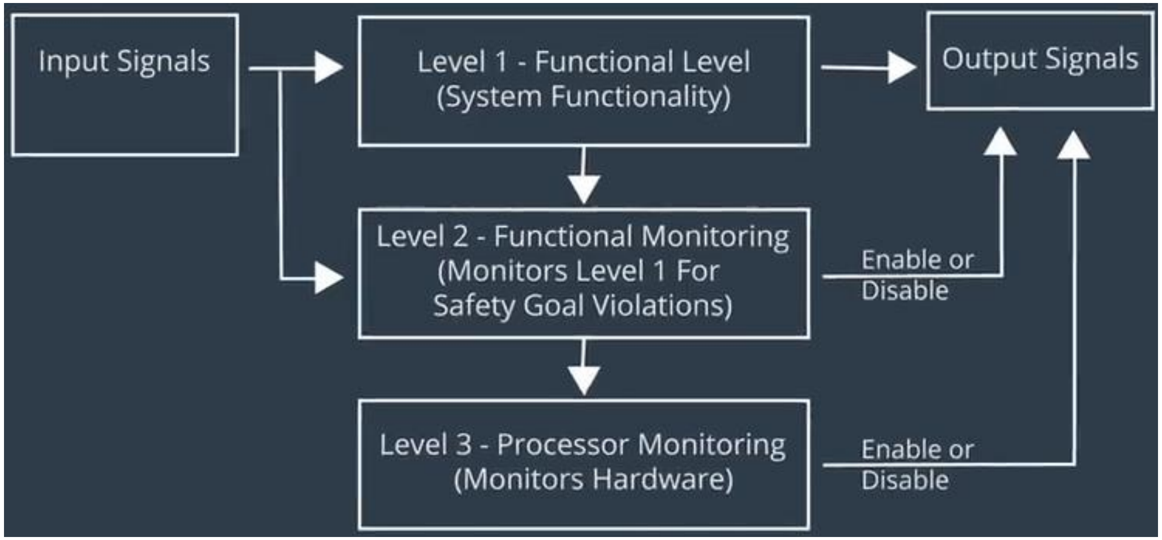
- Communication channels between ECUs often have QM level, hence data transmission errors on the channel are possible.
- E2E protocol can be used to detect errors due to transmission errors.
  - – e.g., with CRCs (Slide 56)

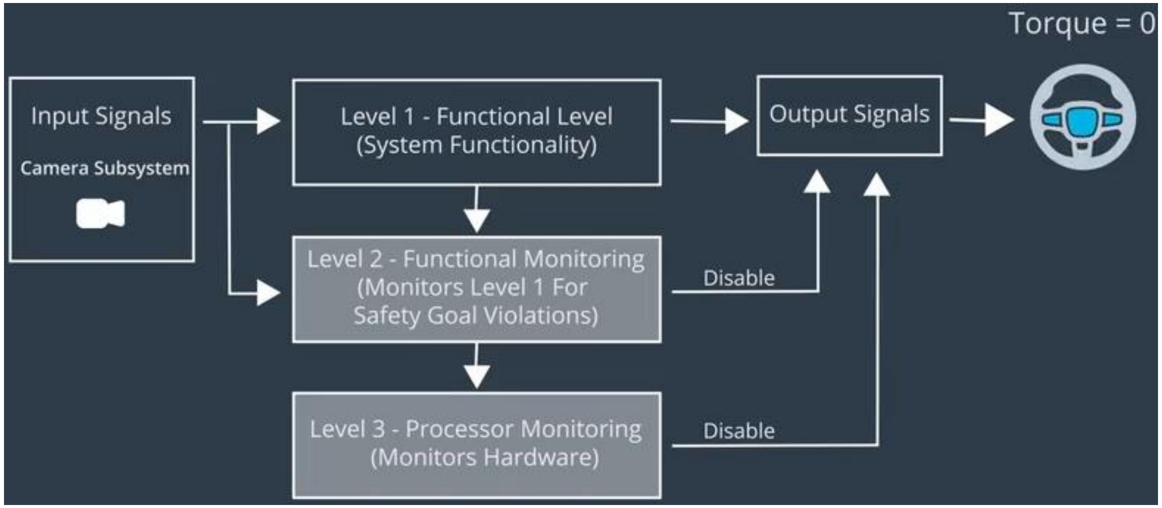




Software Architecture Safetey Design Pattern: E-Gas

- Top: the E-Gas architecture with 2 levels of monitoring for the functional level.
- Bottom: if Level 2 monitoring detects a safety goal violation, output from level 1 is disabled, and Level 2 software leads the system to a safe state by setting Torque = 0.


Input Signals
Level 1 - Functional Level (System Functionality)
Output Signals
Level 2 - Functional Monitoring (Monitors Level 1 For Safety Goal Violations)
Enable or Disable
Level 3 - Processor Monitoring (Monitors Hardware)
Enable or Disable


Torque = 0
Input Signals
Camera Subsystem
Level 1 - Functional Level (System Functionality)
Output Signals
Level 2 - Functional Monitoring (Monitors Level 1 For Safety Goal Violations)
Disable
Level 3 - Processor Monitoring (Monitors Hardware)
Disable