life.augmented

# Functional Safety in Electronic Systems: Principles and Applications

Alessandro Bastoni

Functional Safety Expert

STMicroelectronics

# General diclaimer on Exercitations

All material included/used for exercitations has been prepared for teaching purposes.

Accordingly, accuracy of presented examples is not 100%, because simplifications have been done here and there to boost the focus on specific aspects related to the teached topics. Also the detail level  can vary in different part of the same example, again because of teaching purposes.

WARNING: applications presented  in this document cannot be considered real, accurate use cases. Their mere replication in real projects may lead to mistakes and missing compliance to the safety standards. Use them just for learning activity.

# Exercitations

## Ex #1 – Safety lifecycle example

**Summary:**

- **Safety lifecycle + system design example**

life.augmented

# Our example: cleaning robot

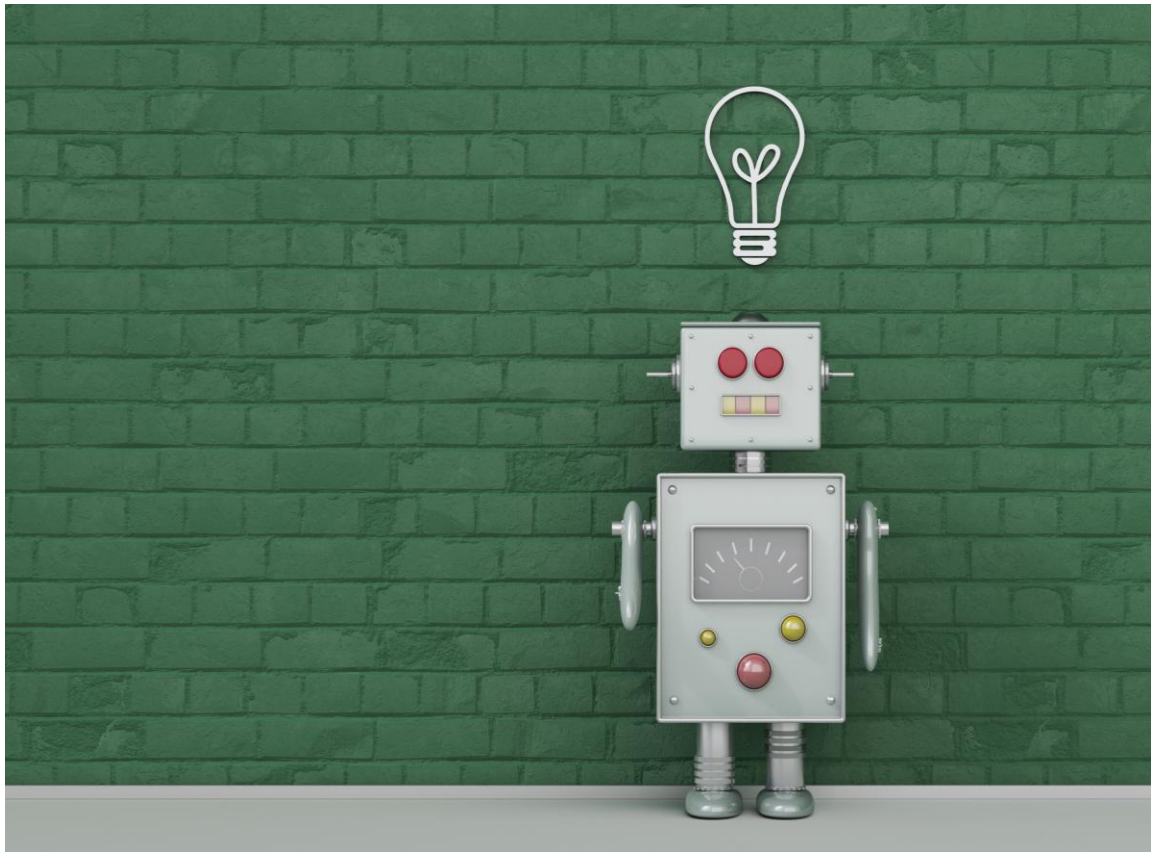
Battery-operated

Autonomous

Trajectory calculation and pattern learning

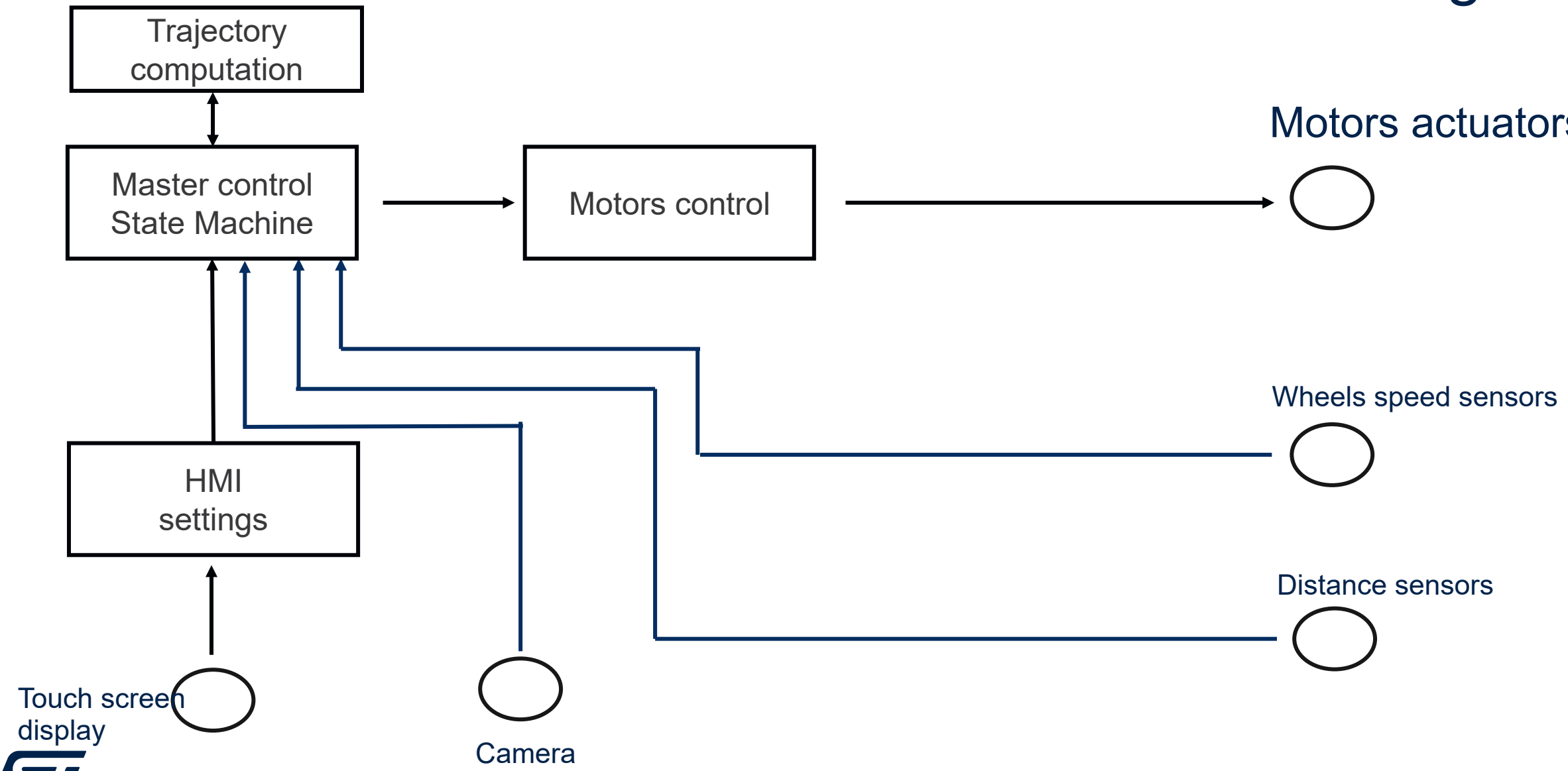
HMI interface via graphic display for user experience

IEC61508 assumed as reference standard



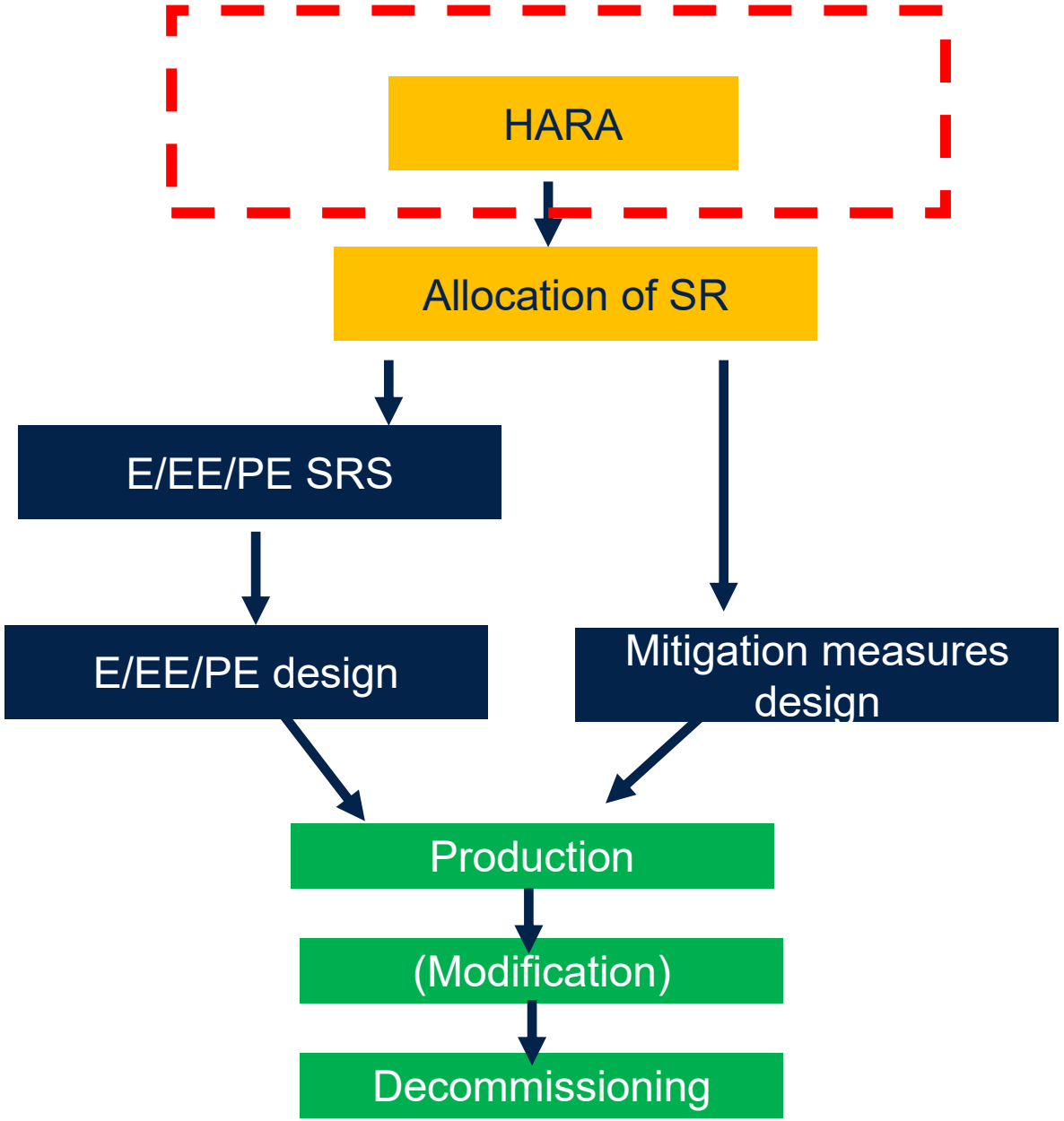
life.augmented

# Functions architecture - original




life.augmented

# Safety lifecycle




Verification

Documentation

Safety Assessment

life.augmented

# HARA example

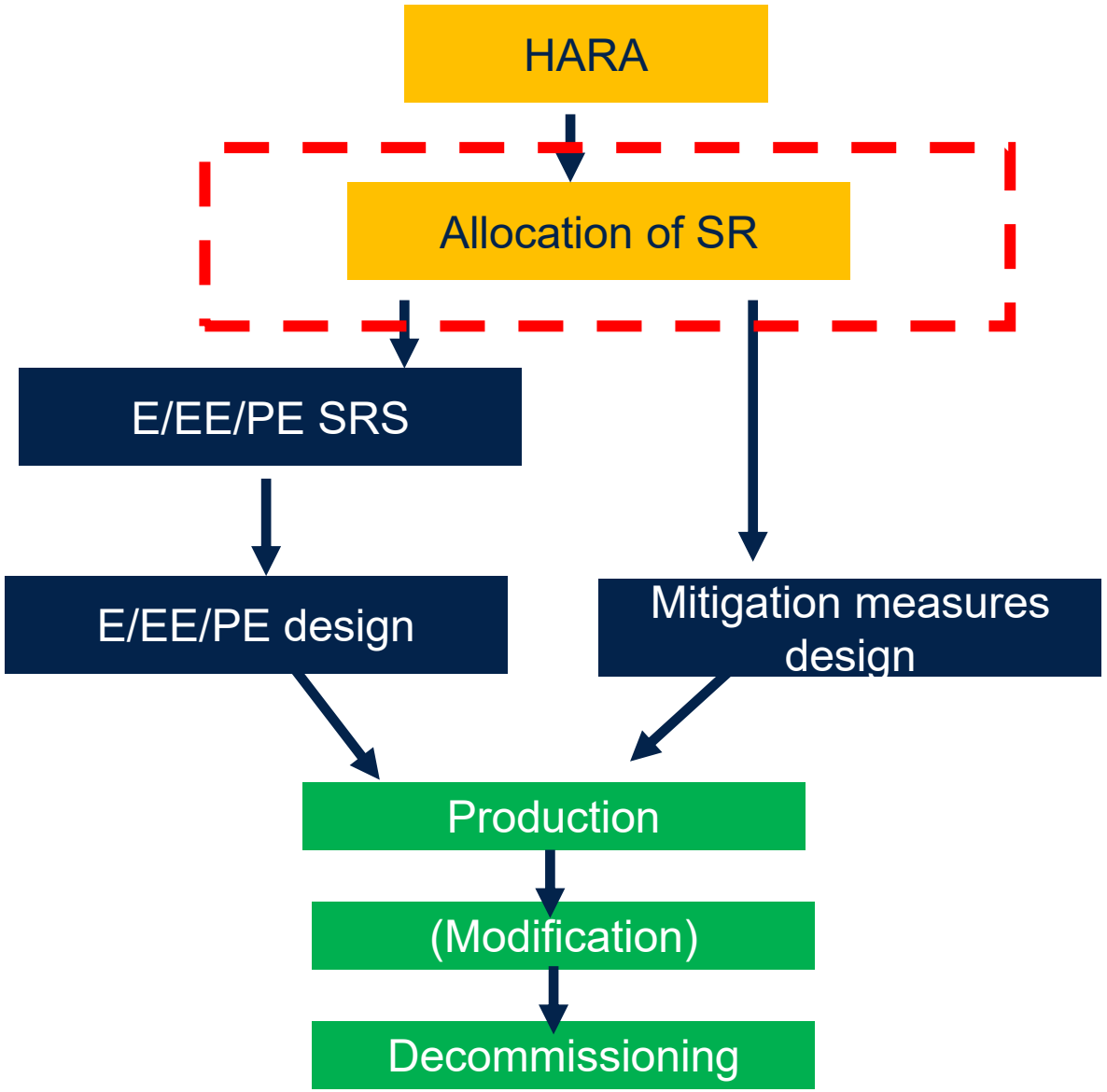
Severity (Sev) 1=low 10=high  Exposure (Exp) 1=improbable 10=frequent

| Hazard | Sev | Exp | R | Safety Function | Sev | Exp | R |
|---|---|---|---|---|---|---|---|
| Robot stuck & and engines on for a certain time, leading to a fire because of overheat | 7 | 7 | 49 | SF1: Avoid robot being stuck with engines on for a time > Tx msec | 7 | 2 | 14 |
| Robot colliding at speed with an object, leading to falling inject potentially injuring people | 4 | 7 | 28 | SF2: Avoid robot front collision with any object at speed > Vx cm/sec | 4 | 2 | 8 |
| Robot falling from a stairs hole or from open floor and hitting people | 6 | 5 | 30 | SF3: Avoid to fall from stairs/open floors | 6 | 2 | 12 |

*Notional example of HARA evaluation*

life.augmented

# Safety lifecycle




Verification

Documentation

Safety Assessment

life.augmented

# Our example: Safety Functions & Safe State

The H&R analysis identifies three coexisting safety functions:

***SF1: Avoid robot being stuck with engines on for a time > Tx msec***

***SF2: Avoid robot front collision with any object at speed > Vx cm/sec***

***SF3: Avoid to fall from stairs/open floors***

The simplified common associated Safe State is <<motors stop + brakes (*) (+ alarm beep and wait for user intervention)>>  (therefore: *de-energize* for all safey functions)

The safety targets are assumed to be SIL2 for hardware and SC2 for software

No other high level safety requirements are defined for the system -> excluding trajectory calculation and HMI from safety scope

(*) In this notional example, braking is assumed to be intrinsic when motor is stopped

# Allocation

***SF1: Avoid robot being stuck with engines on for a time > Tx msec***

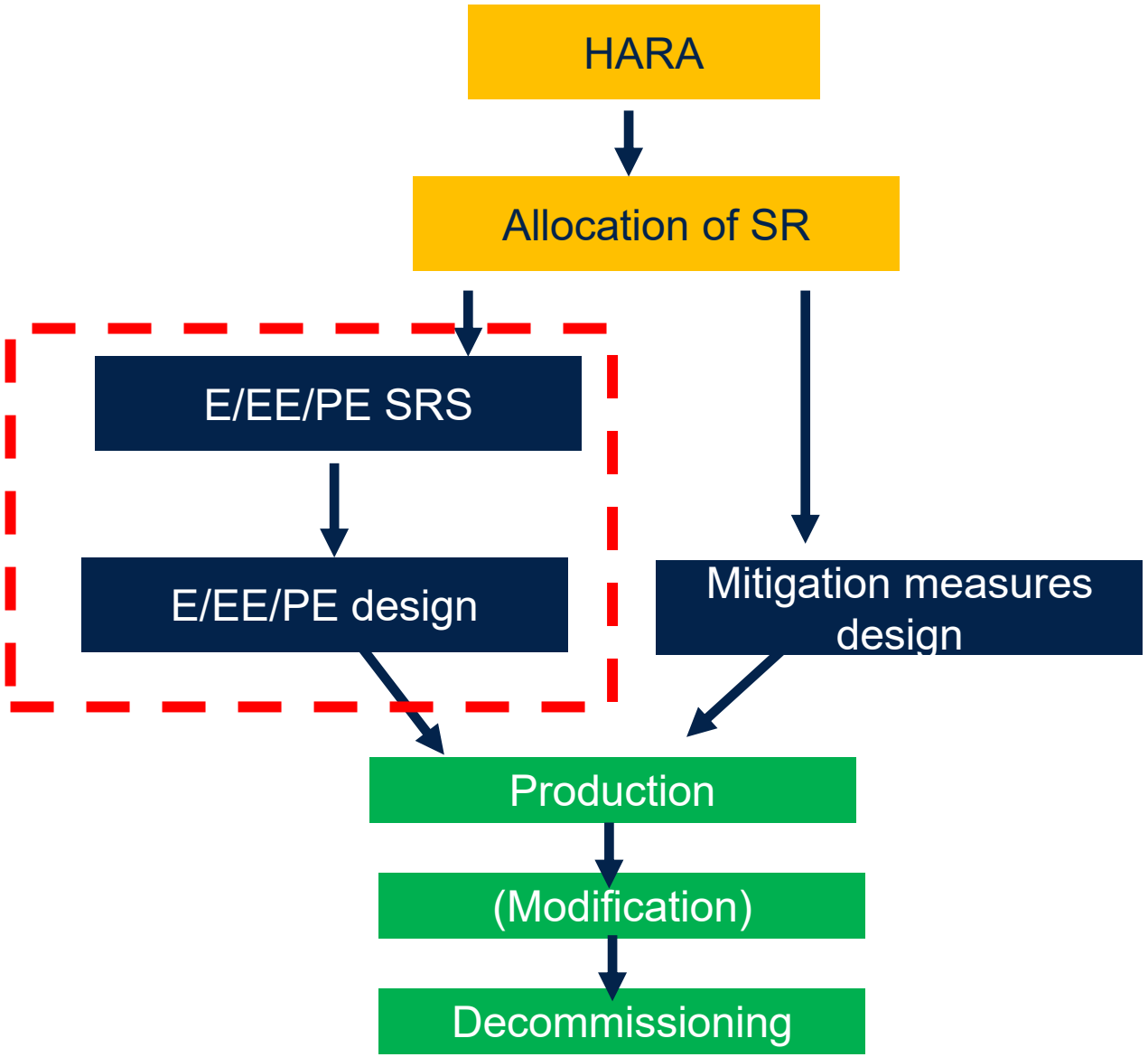
***SF2: Avoid robot front collision with any object at speed > Vx cm/sec***

Allocated to E/EE/PE, hardware and software (see next slides)

***SF3: Avoid to fall from stairs/open floors***

Allocated to system-level mitigation measures e.g. restrict use fo the robot  in environment where the hazard is not present (no stairs…) or add to product description additional measures like barriers

# Safety lifecycle

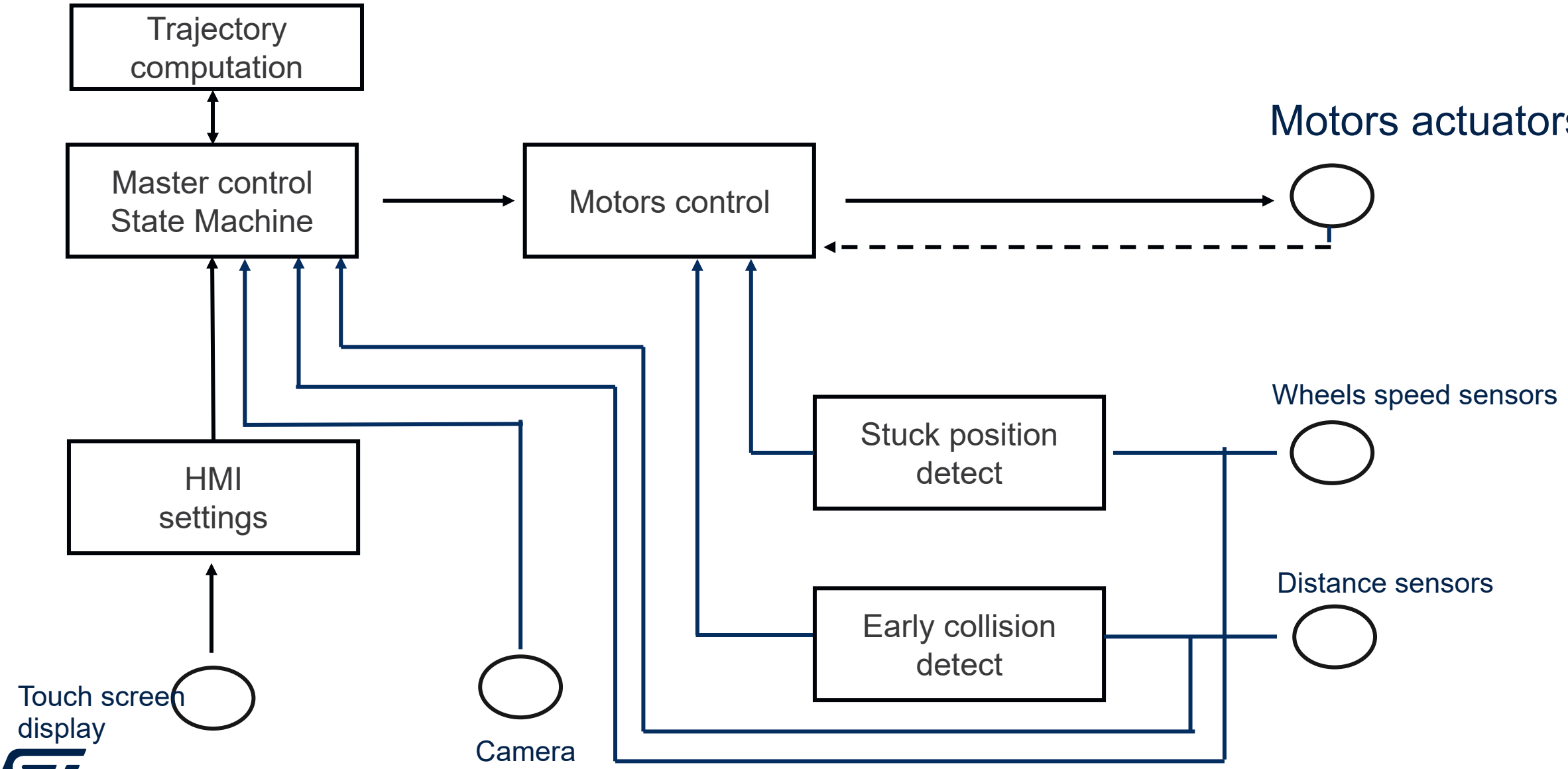



Verification

Documentation

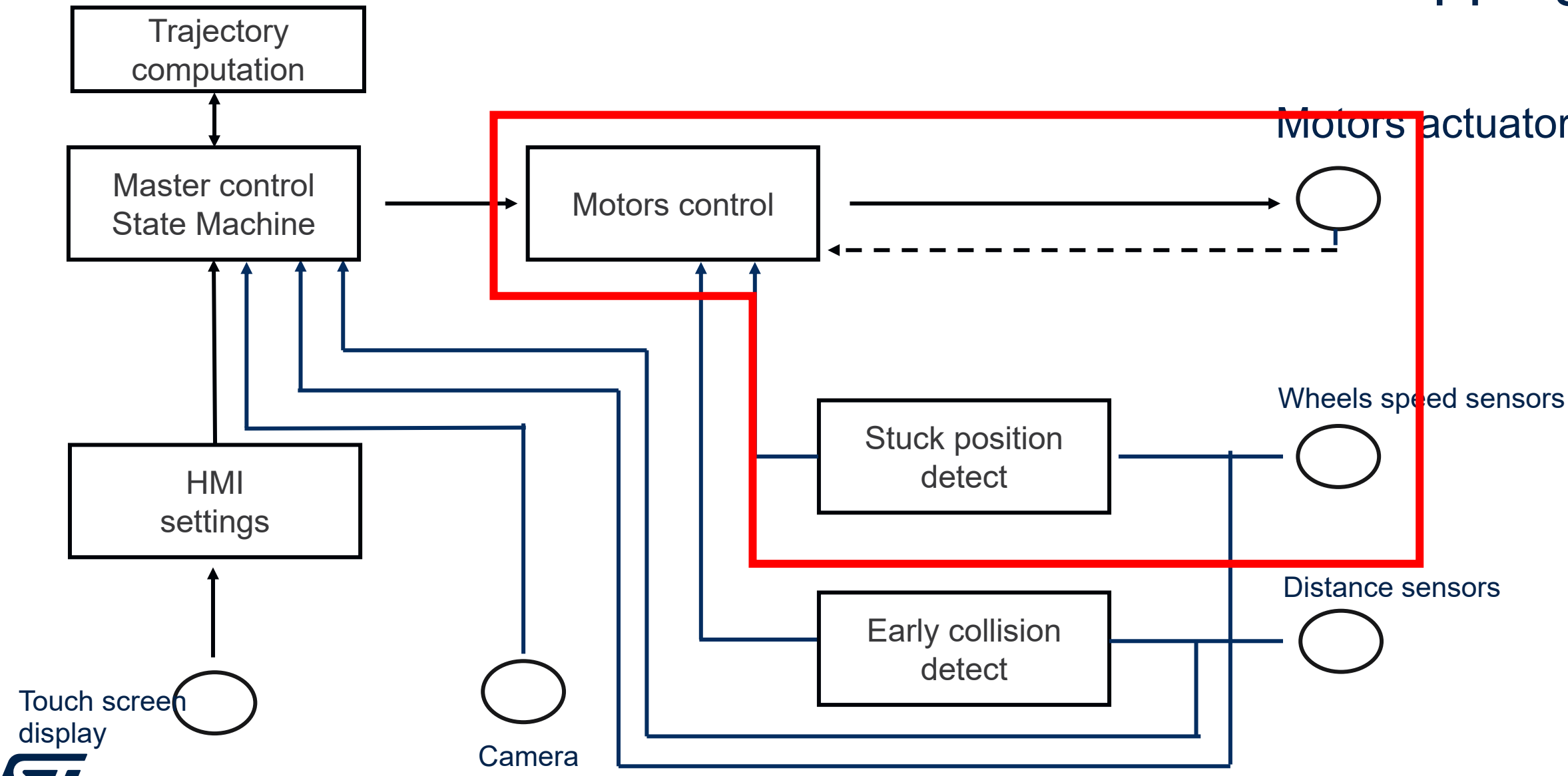
Safety Assessment

# Functions architecture - modified



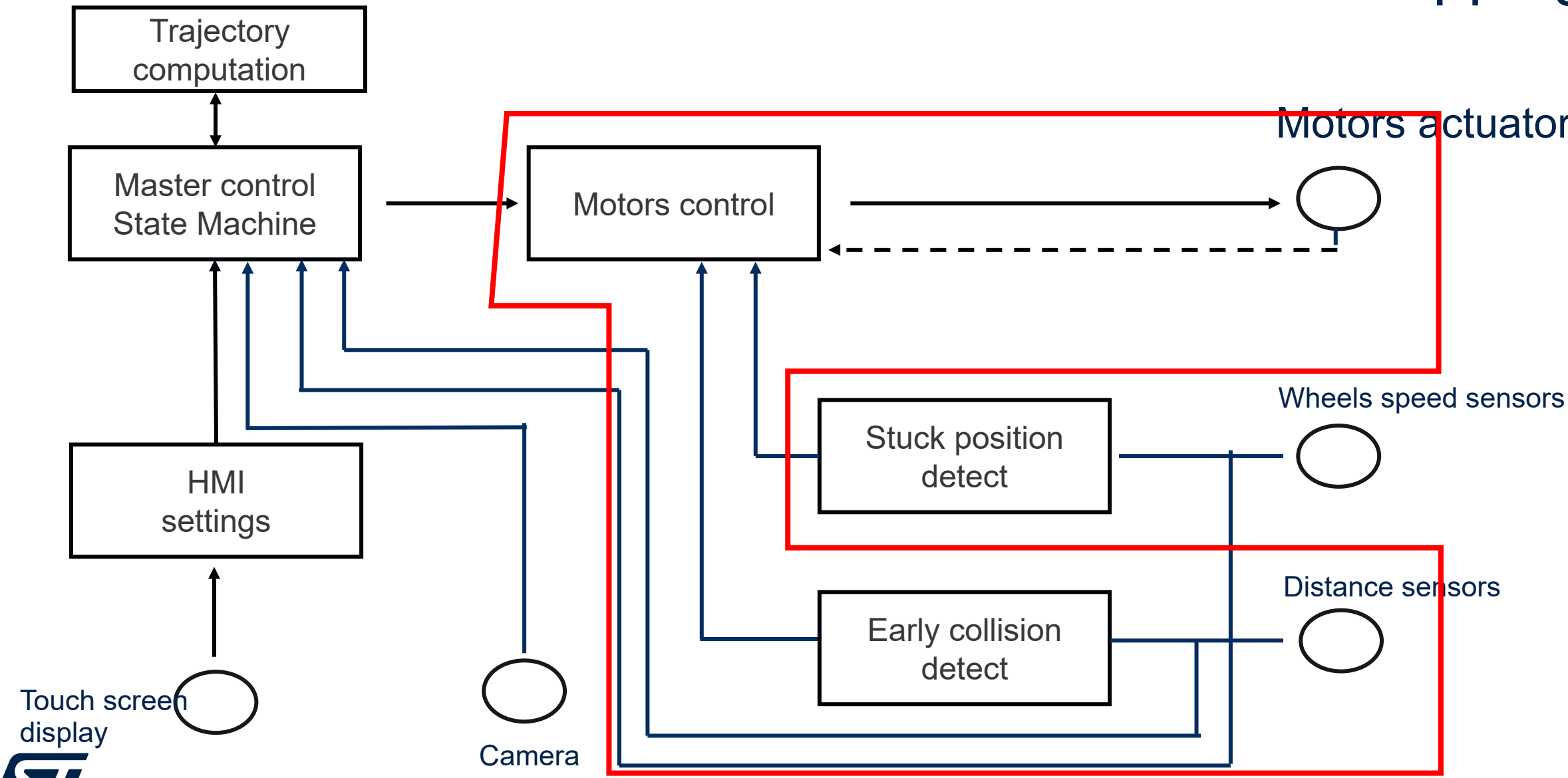

life.augmented

# SF1 mapping

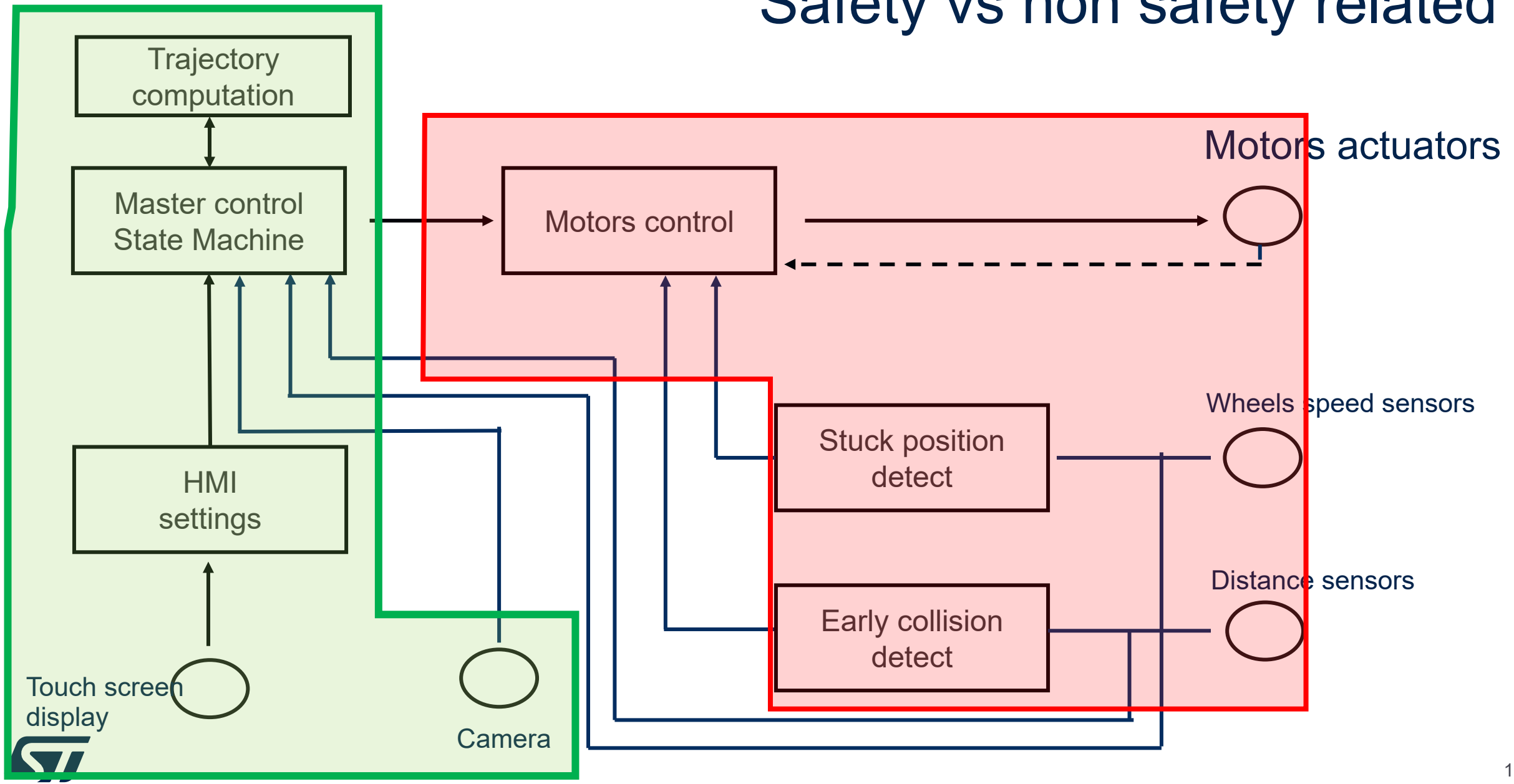



life.augmented

# SF2 mapping


Trajectory computation
Master control State Machine
Motors control
Motors actuators
HMI settings
Stuck position detect
Wheels speed sensors
Early collision detect
Distance sensors
Touch screen display
Camera

# Safety vs non safety related



# Exercitations

## Ex #2 – ASIL decomposition

**Summary:**

- **ASIL decomposition**
- **Design of a subpart of the system**

life.augmented

# ASIL decomposition – example 1/6

Consider this example: a system with an actuator that is triggered on demand by the driver using a touch panel interface on the dashboard. The actuator provides a comfort function (e.g. roof opening on cabriolet car) if the vehicle is at zero speed, but can cause hazards if activated above 20 km/h.

Initial architecture is composed by:

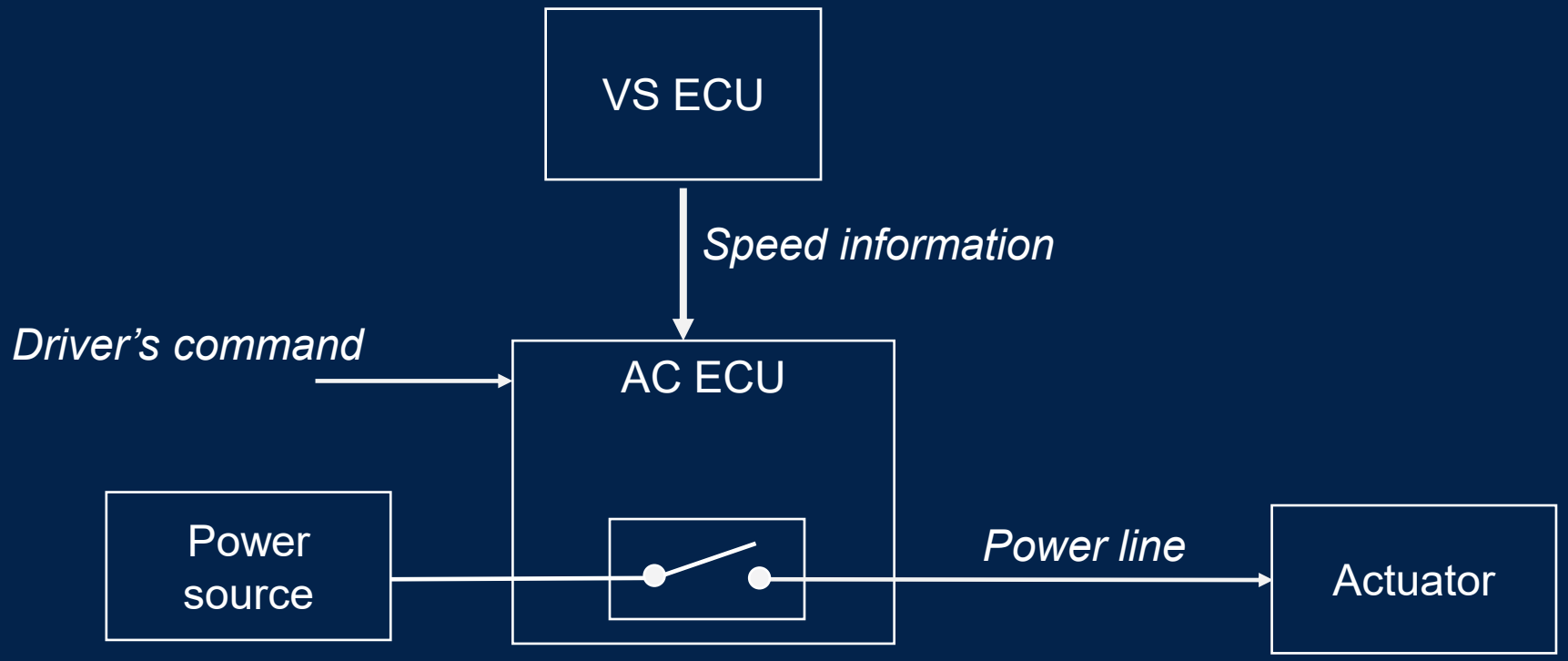
The touch panel input is read by a dedicated ECU (referred to as "Actuator Control ECU (AC ECU)" ), which powers the actuator through a dedicated power line.

The vehicle is also fitted with an ECU which is able to provide the vehicle speed by dedicated messages. The ability of this ECU to provide the information that the vehicle speed is greater than 20 km/h is assumed to be compliant with ASIL C requirements. This ECU is referred to as "VS ECU".

It is assumed that the HARA indicates an ASIL C classification for the hazardous event. So related safety goal inherits ASIL C.

# ASIL decomposition – example 2/6

Initial architecture



# ASIL decomposition – example 3/6

The associated safety goal is

Safety Goal 1: Avoid activating the actuator while the vehicle speed is greater than 20 km/h: ASIL C

Resulting Functional Safety Concept is the following:

Requirement A1: The VS ECU provides accurate vehicle speed information to the AC ECU. → ASIL C

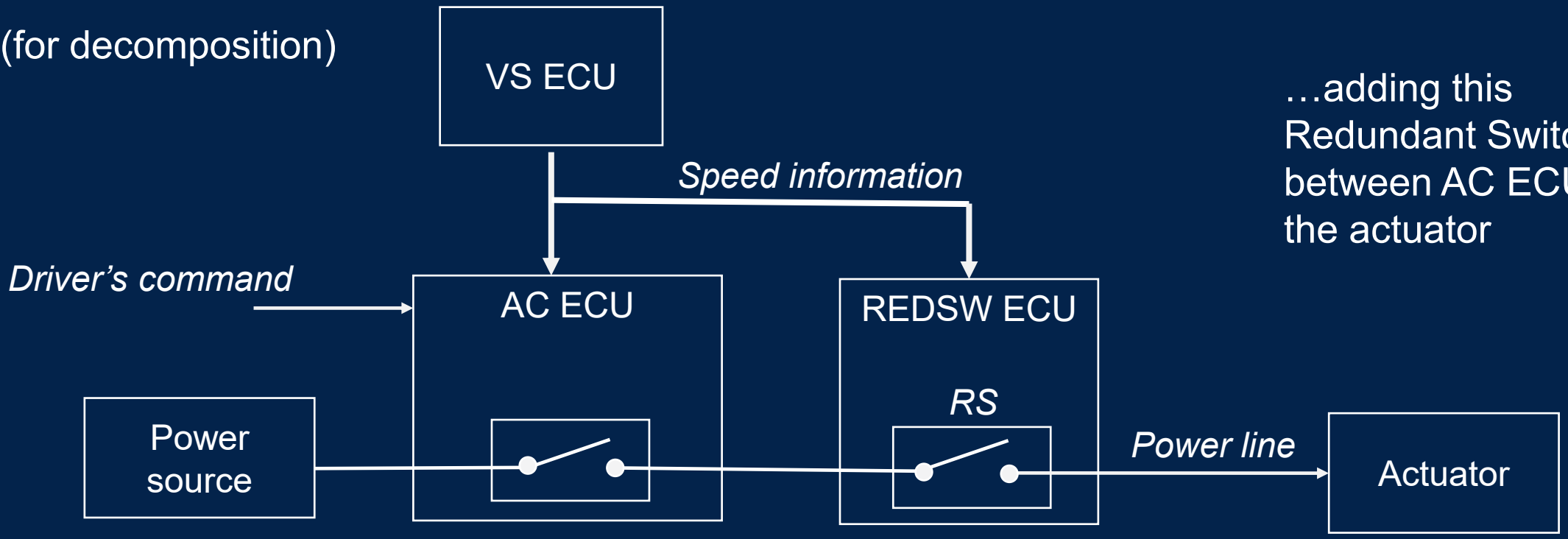
Requirement A2: The AC ECU does not power the actuator if the vehicle speed is greater than 20 km/h. → ASIL C

Requirement A3: The actuator is activated only when powered by the AC ECU. → ASIL C

# ASIL decomposition – example 4/6

Modified architecture

(for decomposition)




…adding this Redundant Switch ECU between AC ECU and the actuator

# ASIL decomposition – example 5/6

Functional Safety requirements after the modification:

Requirement B1: The VS ECU provides accurate vehicle speed information to the AC ECU. → ASIL C (unchanged)

Requirement B2: The AC ECU does not power the actuator if the vehicle speed is greater than 20 km/h. → ASIL X(C)

Requirement B3: The VS ECU provides accurate vehicle speed information to the REDSW ECU. → ASIL C

Requirement B4: The RS switch in REDSW ECU is in an open state if the vehicle speed is greater than 20 km/h. → ASIL Y(C)

Requirement B5: The actuator is activated only when powered by the AC ECU and the RS switch in REDSW ECU is closed. → ASIL C

Requirement B6: Sufficient independence of the AC ECU and the REDSW ECU is shown. → ASIL C

# ASIL decomposition – example 6/6

According to ASIL decomposition rules, multiple combination are possible:

| # | Requirement B2: ASIL X(C) | Requirement B4: ASIL Y(C) |
|---|---|---|
| 1 | ASIL C(C) requirements | QM(C) requirements |
| 2 | ASIL B(C) requirements | ASIL A(C) requirements |
| 3 | ASIL A(C) requirements | ASIL B(C) requirements |
| 4 | QM(C) requirements | ASIL C(C) requirements |

BUT because complexity for B2 >> B4 (*), most reasonable decompositions are options #4 and #3

*(*) in reality, this should be a conclusion after a safety analysis e.g. FTA*

# Safe state(s)

Recall the Safety Goal: Avoid activating the actuator while the vehicle speed is greater than 20 km/h

Looking to Req B5: “The actuator is activated only when powered by the AC ECU and the RS switch in REDSW ECU is closed.” we can state that :

Safe State =  (AC ECU does not generate power) and/or (RS switch is open)

The local safe state for REDSW ECU is then: “RS switch is open”. This is a *degraded mode* for the system (the comfort function can no longer be actuated upon driver request even when vehicle speed is lower than 20km/h. But safety is achieved because risk is mitigated.

# Exploding FSR into TSR

Requirement B3: The VS ECU provides accurate vehicle speed information to the REDSW ECU.

TSR3.1: REDSW ECU shall correctly acquire vehicle speed distributed by VS ECU

Requirement B4: The RS switch in REDSW ECU is in an open state if the vehicle speed is greater than 20 km/h.

TSR4.1: REDSW ECU shall open RS switch if vehicle speed is > 20km/h

TSR4.2: REDSW ECU shall open RS switch if correct vehicle speed is not available

# Exploding FSR into TSR

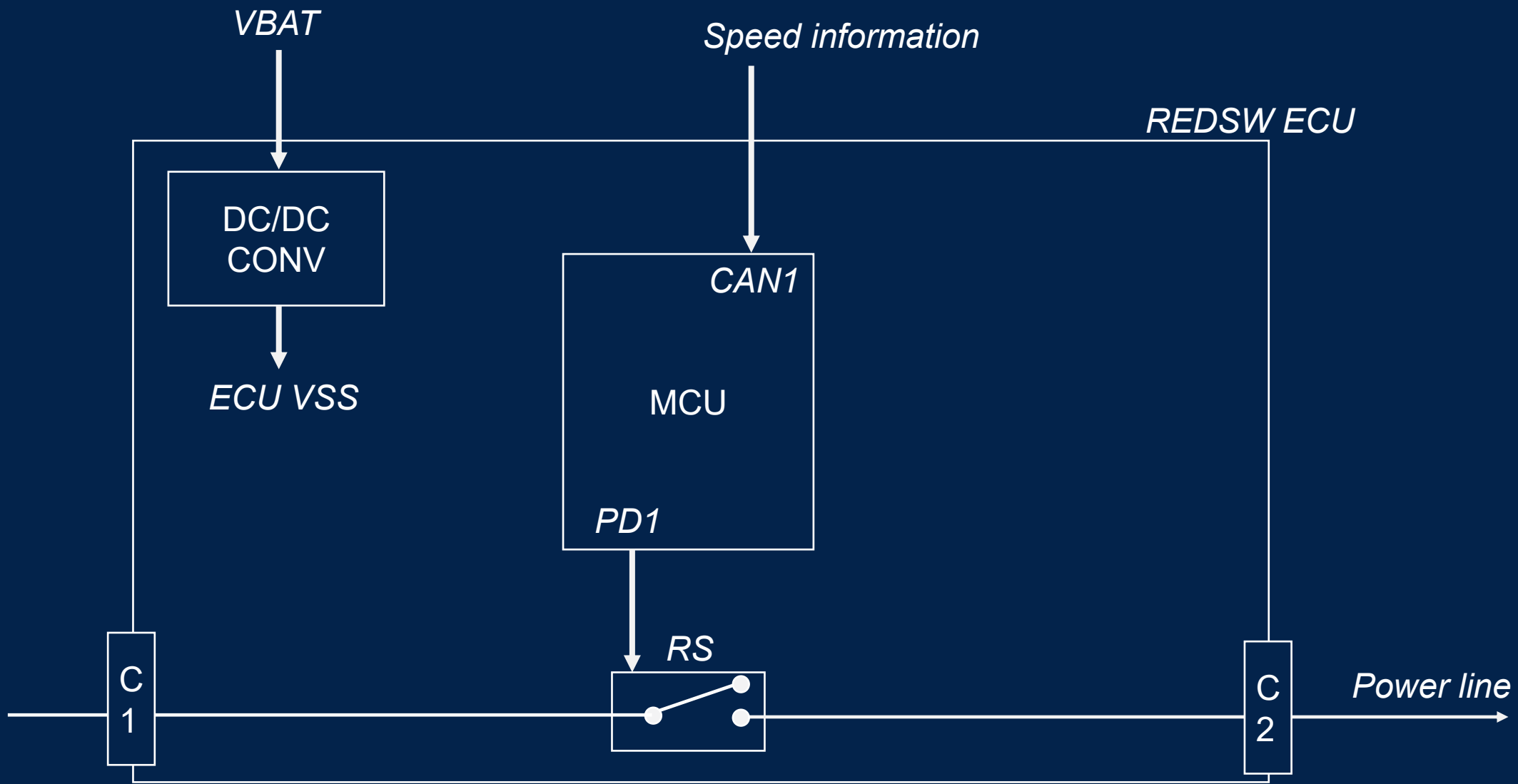
Requirement B5: The actuator is activated only when powered by the AC ECU and the RS switch in REDSW ECU is closed.

- TSR5.1: REDSW RS shall be connected in series to the power line between the AC ECU and the actuator
- TSR5.2: REDSW RS shall connect the power line from AC ECU to the actuator through the switch RS
- TSR5.3: REDSW RS shall not generate direct supply for the actuator

Requirement B6: Sufficient independence of the AC ECU and the REDSW ECU is shown.

- TSR6.1: REDSW ECU shall generate its local power supply from vehicle VBAT
- TSR6.2: REDSW ECU shall receive vehicle speed directly from VS ECU
- TSR6.3: REDSW ECU shall not share signals or information with AC ECU
- TSR6.3: REDSW ECU shall be able to switch the system in safe state without AC ECU intervention

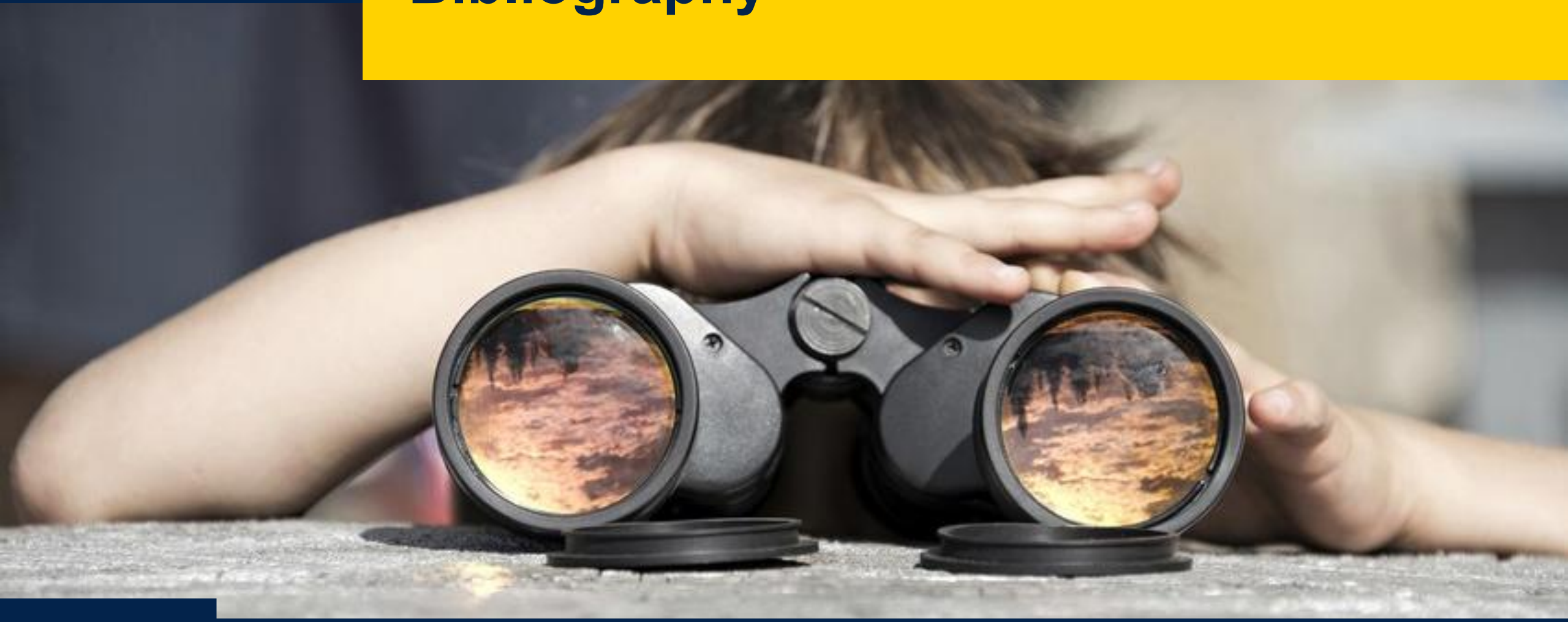
# REDSW ECU implementation



# Backup slides

life.augmented

# Bibliography



life.augmented

# Thank you



life.augmented