

Extending the Shell Model via Cause/Consequence Analysis of Component Failures

A Method for systematic
System and Failure
Analysis

Authors:
W. Sebron
H. Tschürtz
P. Krebs

Overview

- > Motivation & Background
- > Extending the Shell Model
- > Short Example
- > Conclusion

The Initial Problem

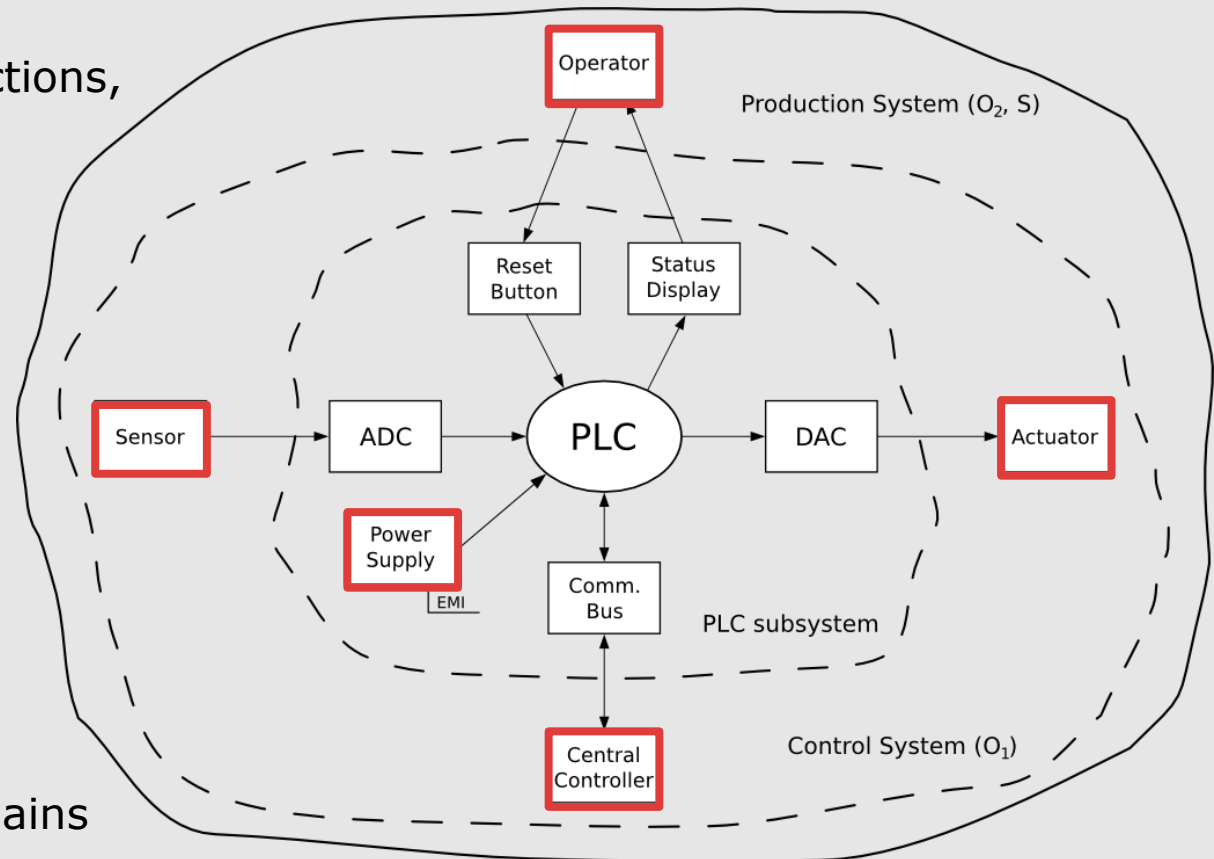
- > Shell Model as a start for aiding structured system definition and delineation
- > Missing link between Shell Model Analysis and further (traditional?) analysis methods
- > Need for a structured, methodical transition of outcomes of the Shell Model Analysis to next step deeper analysis
- > Early lifecycle problem, no systematic method yet available

Overview

- > Motivation & Background
- > Extending the Shell Model
- > Short Example
- > Conclusion

FMC – Derivation of Top-Level Functions

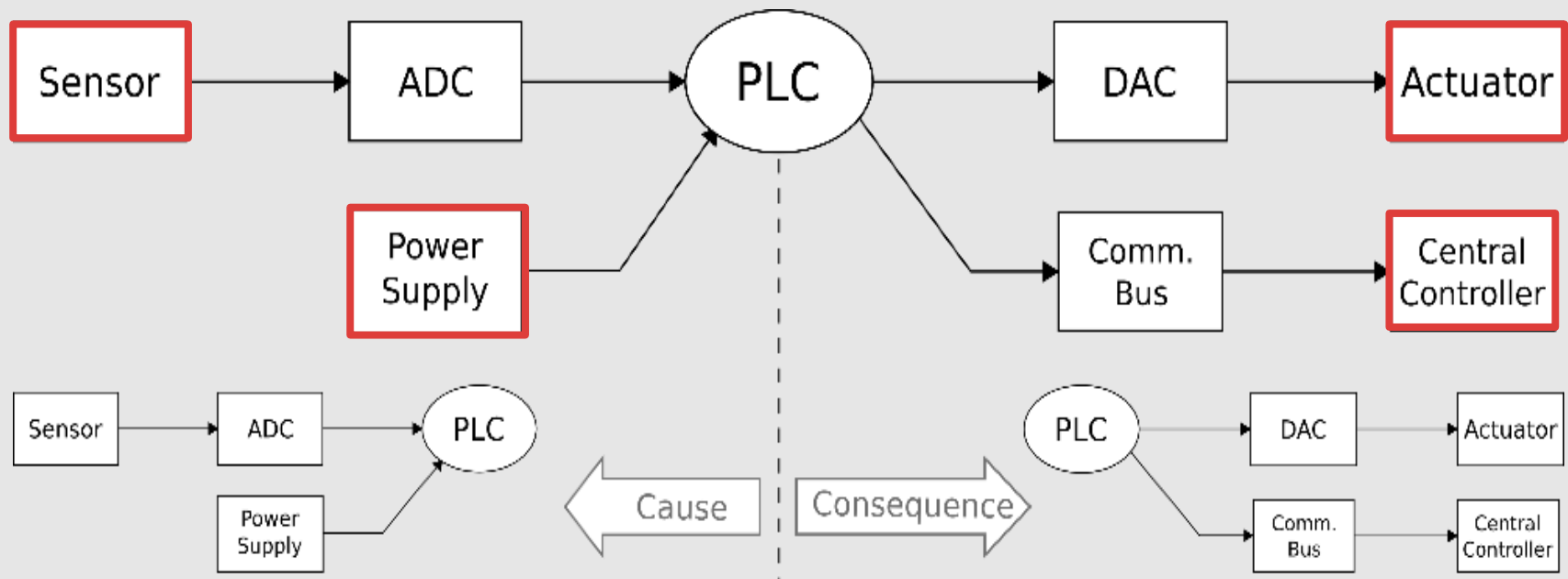
- > For the top-level functions, logical sequences of the relation chains have to be identified
- > Typically, some input component is taken as starting point and some output component as ending point for a function
- > These logical chains are called Function-Message-Chains (FMC)



SUC (System Under Consideration) ... in this example, the PLC (Programmable Logic Controller)
FMC ... Function-Message-Chain

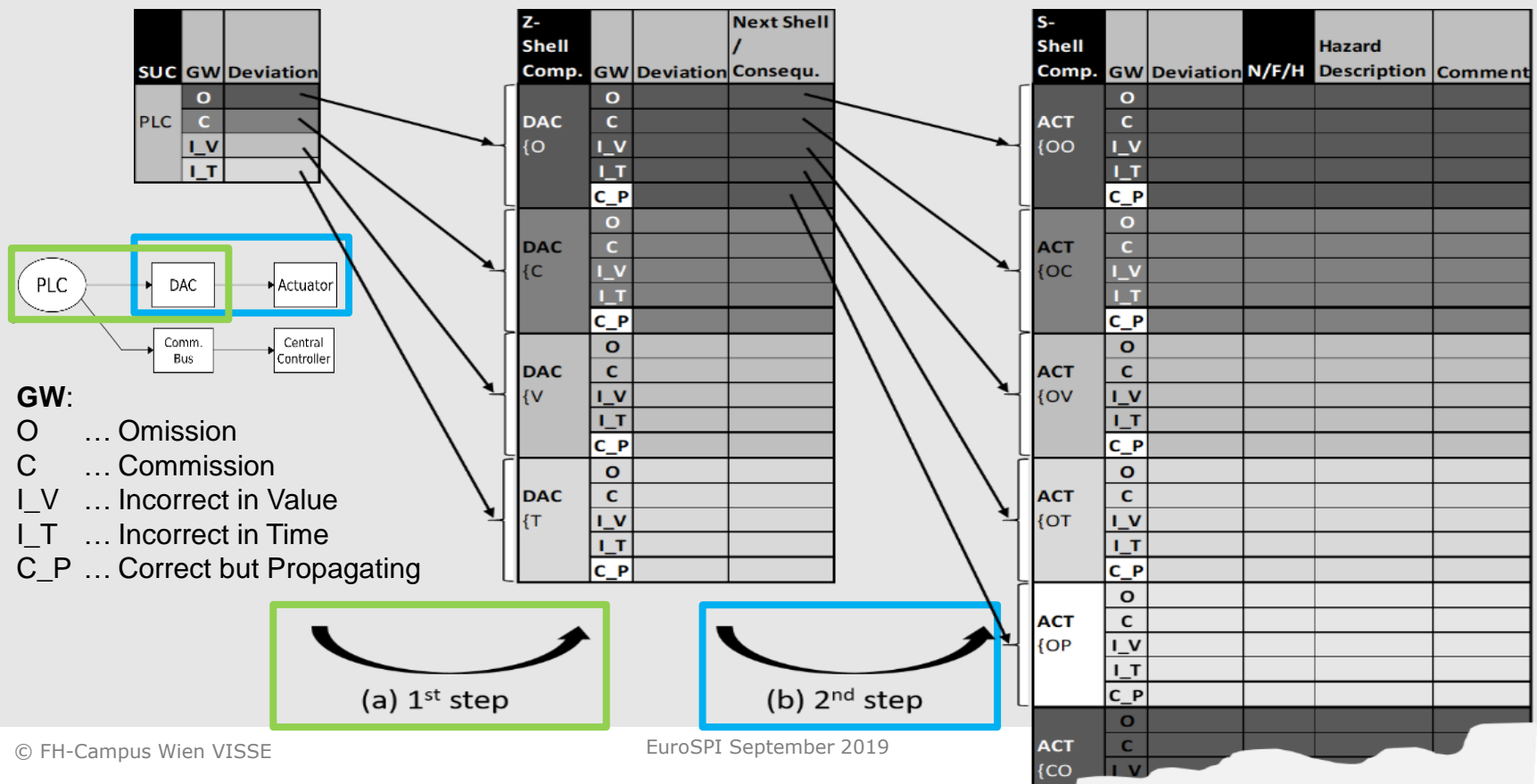
Splitting up the FMC of a Function

> One specific FMC of the previous Shell Model Graph:



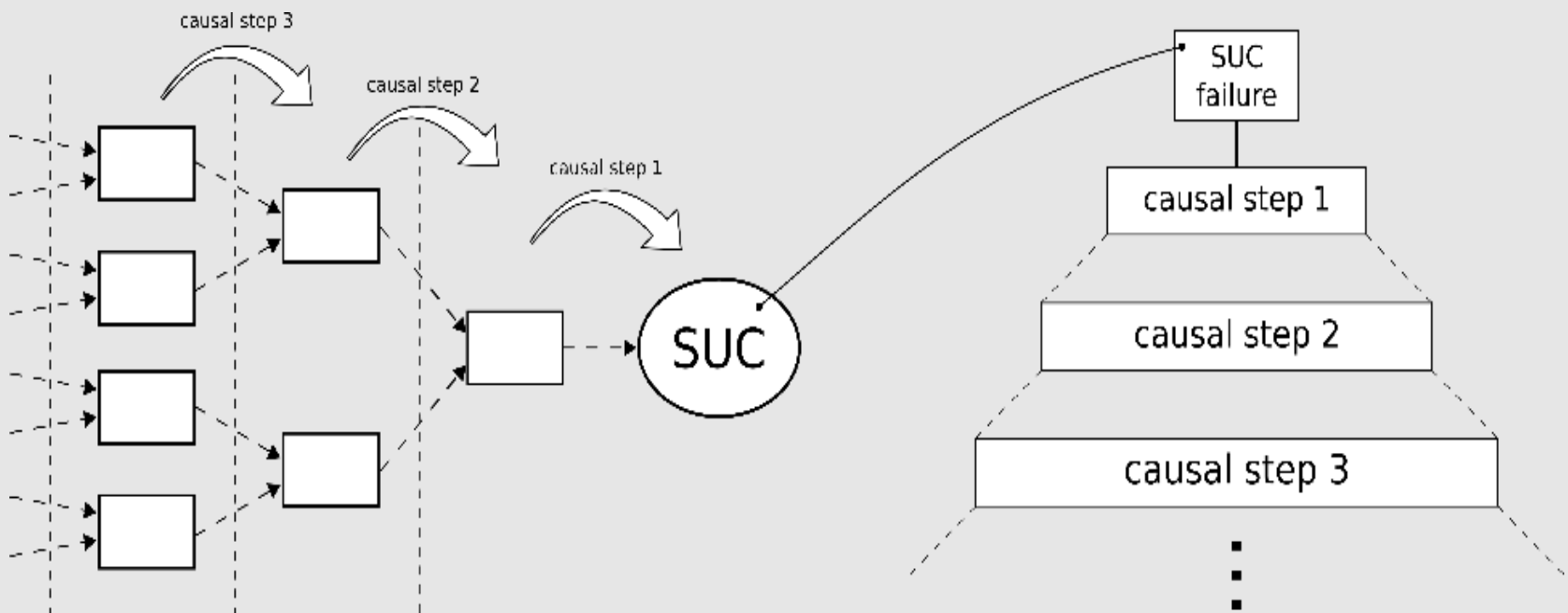
Analysing Consequences via Failure Modes and Effects Analysis (FMEA) using FFA guidewords (GW)

> Consequence analysis steps executed via adapted FMEA and use of specific FFA guidewords (GW)



Analysing Causes via Fault Tree Analysis (FTA)

> Relationship between cause part (left)
and derived Fault Tree (right)

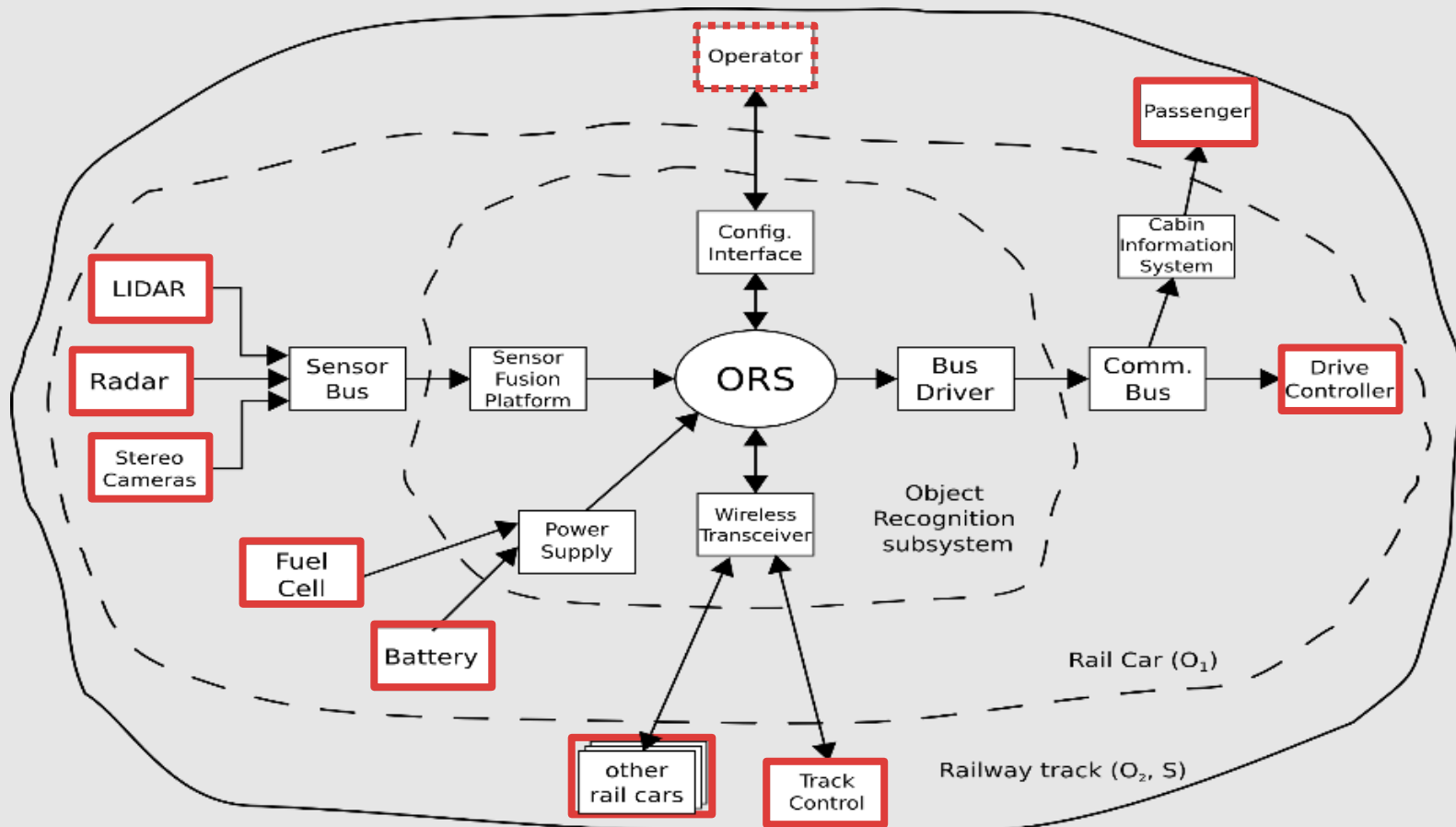


SUC ... System Under Consideration

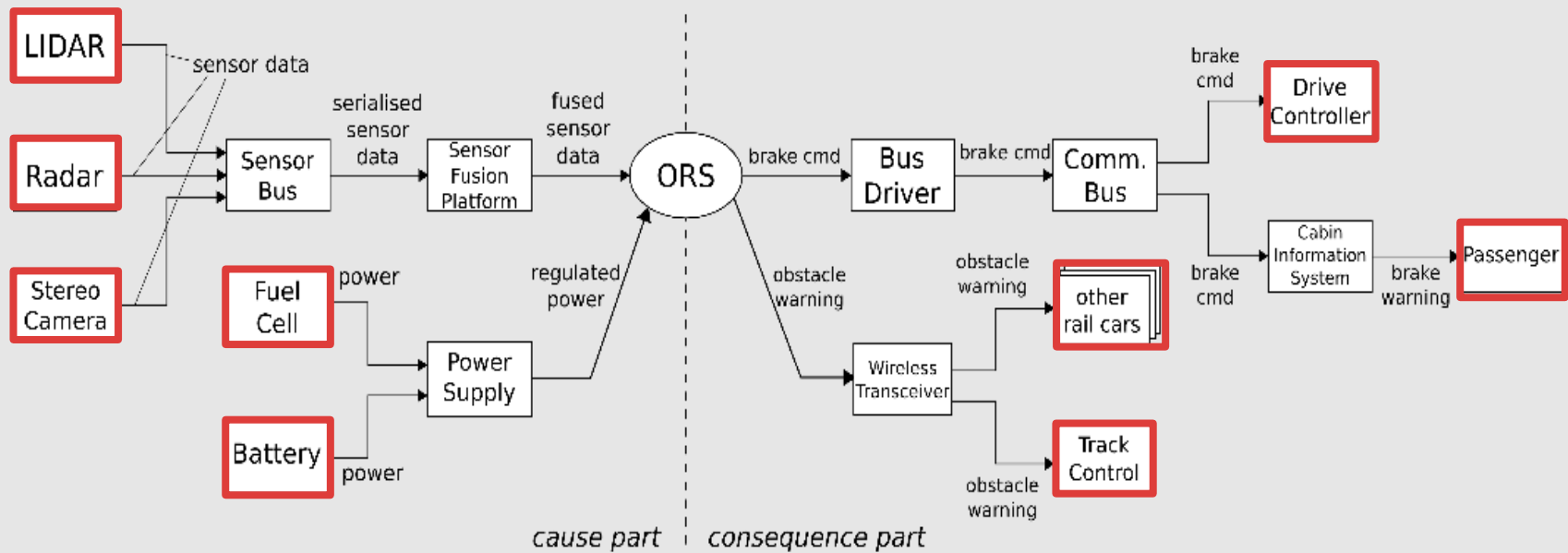
Overview

- > Motivation & Background
- > Extending the Shell Model
- > **Short Example**
- > Conclusion

Shell model of the ORS example



FMC for ORS system function “autonomous emergency braking”

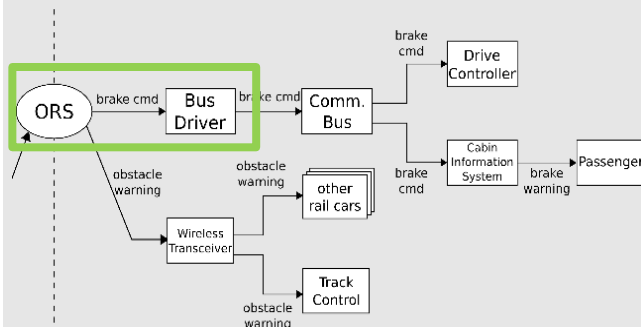


First step of the adapted FMEA according to the FMC

FUN: EMERGENCY FULL BRAKING IS ACTIVATED

SUC	GW	Deviation
ORS	O	no brk cmd out though should
	C	brk cmd out though should not
	I_V	wrong brk cmd out
	I_T	brk cmd out too late

O ... Omission
C ... Commission
I_V ... Incorrect in Value
I_T ... Incorrect in Time
C_P ... Correct but Propagating



Z-Shell Comp.	GW	Deviation	Next Shell / Consequ.
BDR {O}	O	n/a (similar to P)	n/a
	C	n/a	n/a
	I_V	n/a	n/a
	I_T	n/a	n/a
BDR {C}	C_P	no brk cmd on bus though should	O propagated
	O	no brk cmd on bus though should	no cmd {C}
	C	n/a (similar to P)	n/a
	I_V	wrong brk cmd on bus	cmd V changed
BDR {V}	I_T	brk cmd on bus too late	C delayed
	C_P	brk cmd on bus though should not	C propagated
	O	no brk cmd on bus though should	no cmd {V}
	C	n/a	n/a
BDR {T}	I_V	n/a (similar to P)	n/a
	I_T	wrong brk cmd on bus too late	wrong cmd even delayed
	C_P	wrong brk cmd on bus	V propagated
	O	no brk cmd on bus though should	no cmd {T}
BDR {T}	C	n/a	n/a
	I_V	wrong brk cmd on bus	V of delayed cmd changed
	I_T	brk cmd on bus too late	cmd even more delayed
	C_P	brk cmd on bus too late	T propagated

Third step of the adapted FMEA for the running example

n.Shell	Comp.	GW	Deviation	Next Shell	Consequ.
COM {VT}	O		no brk cmd sent though should	no cmd {VT}	
	C		n/a	n/a	
	I_V		n/a (similar to P)	n/a	
	I_T		wrong brk cmd on bus too late	wrong cmd again delayed	
	C_P		wrong brk cmd sent late	VT propagated	

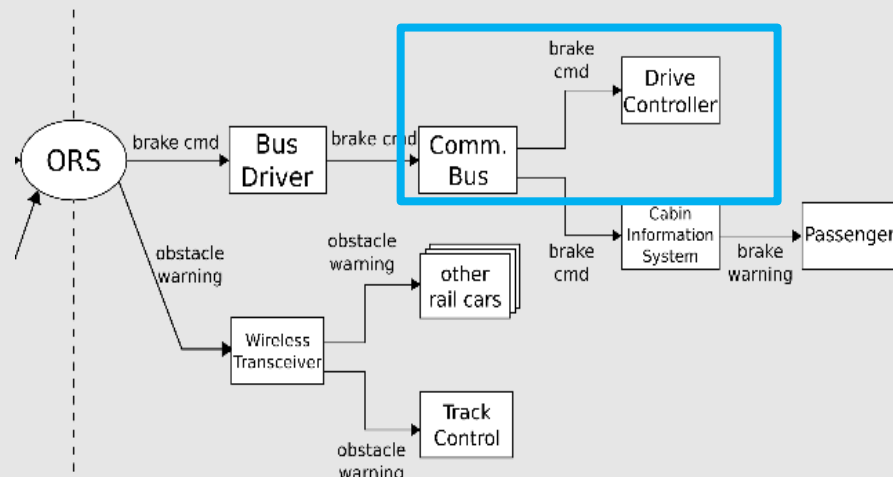
s.Shell	Comp.	GW	Deviation	N/F/H	Hazard	Description	Comment
DRC {VTT}	O		brakes remain inactive	H	Railcar does not brake at all		will hit recognised object
	C		n/a	N	n/a		n/a
	I_V		n/a (similar to P)	N	n/a		n/a
	I_T		brakes are engaged even later, with wrong force	H	Railcar does not stand still in time		will hit recognised object
	C_P		brakes are engaged, but late with wrong force	H	Railcar does not stand still in time		will hit recognised object

GW:

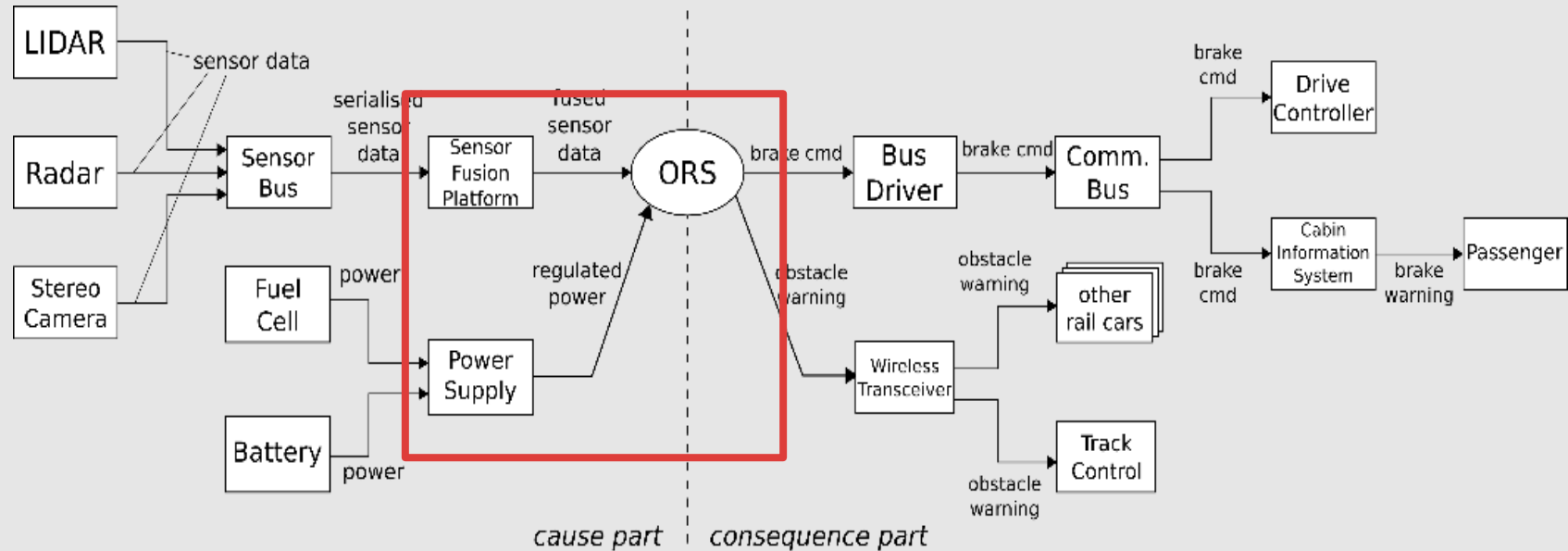
O ... Omission
C ... Commission
I_V ... Incorrect in Value
I_T ... Incorrect in Time
C_P ... Command Propagated

N/F/H:

N ... No Failure
F ... Failure
H ... Hazard

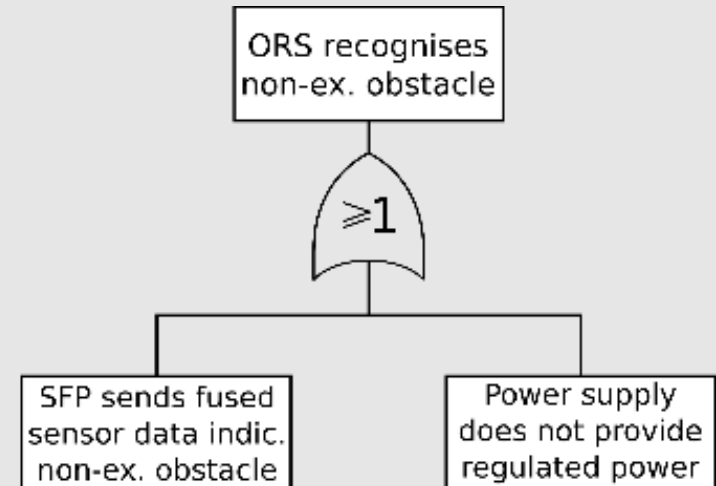
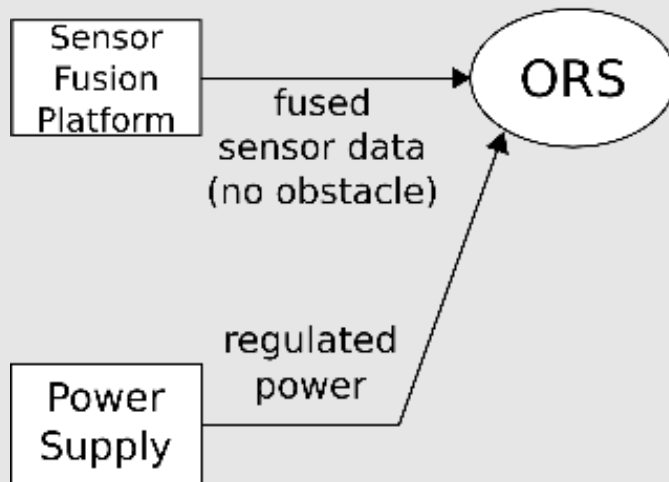


FMC for ORS system function “autonomous emergency braking”

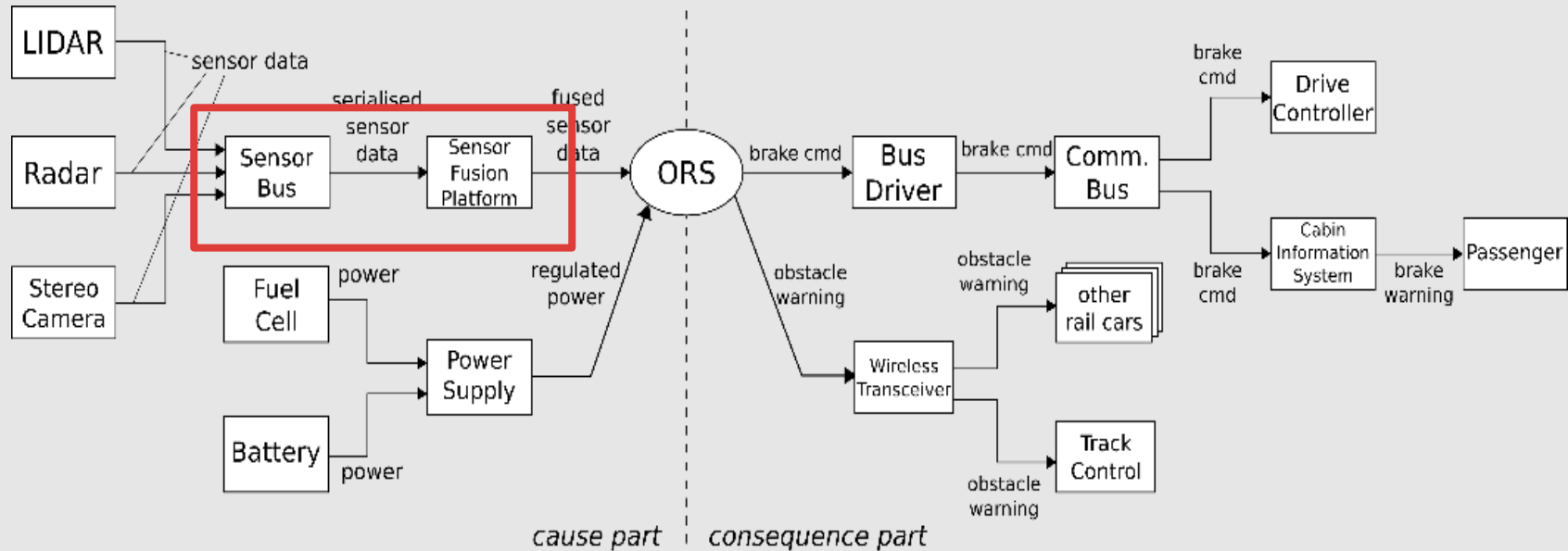


ORS ... Object Recognition System

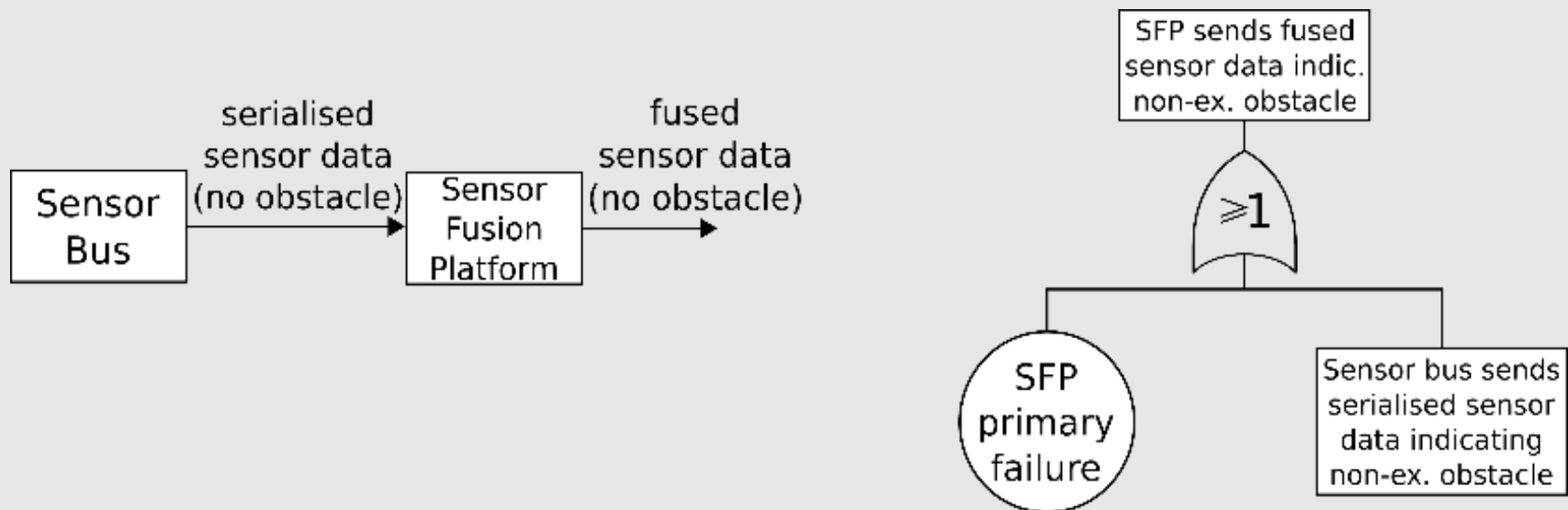
Cause part fragment (left) and derived partial FT (right) from first causal step



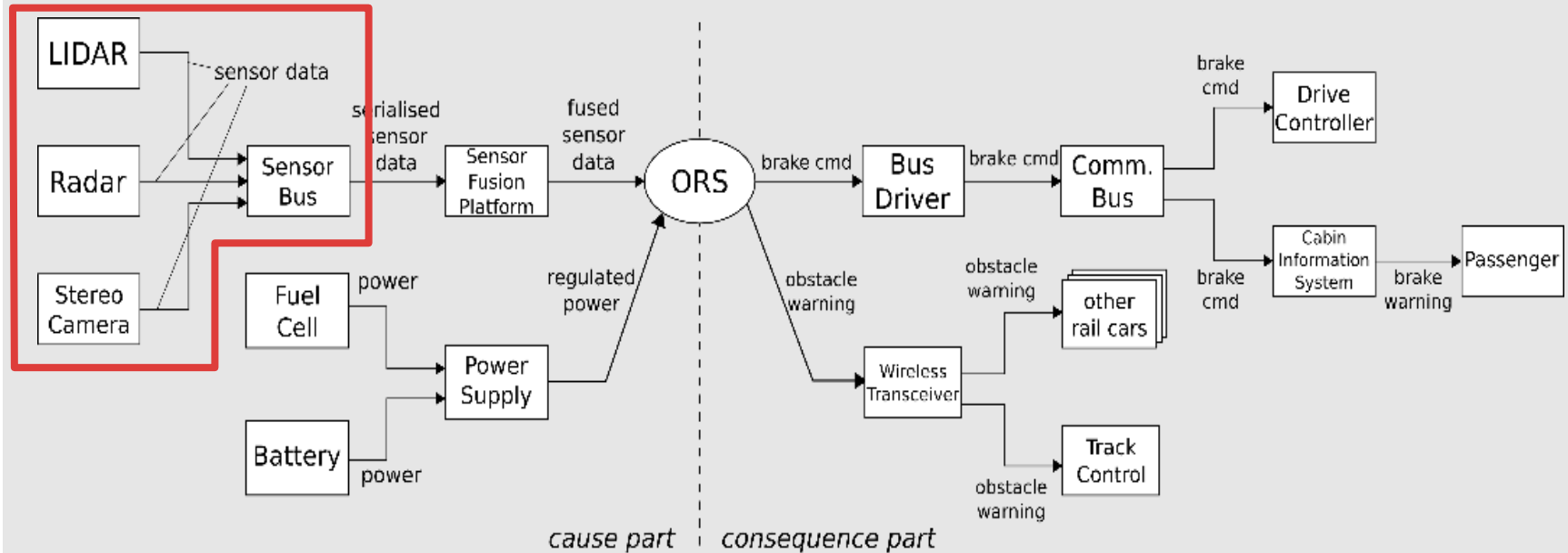
FMC for ORS system function “autonomous emergency braking”



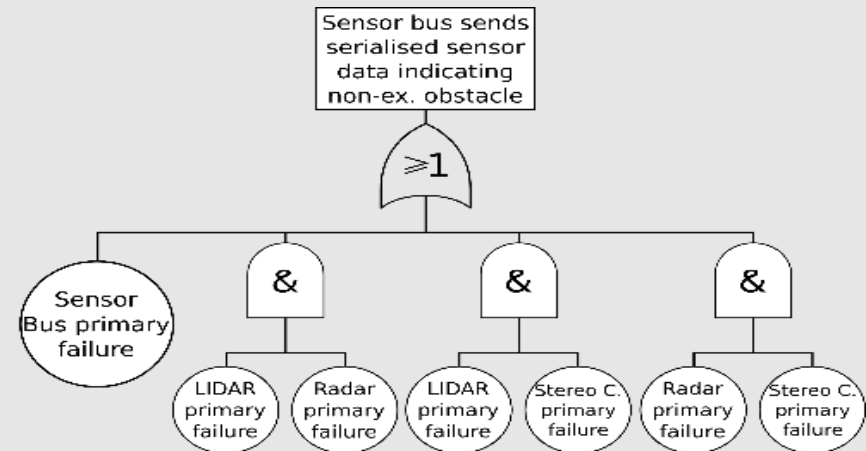
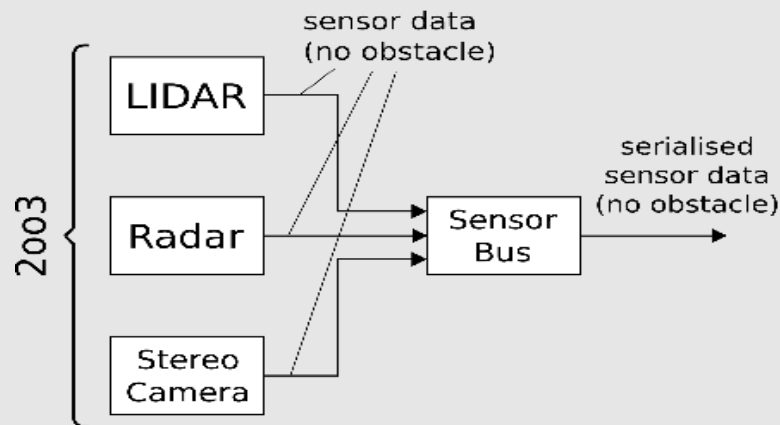
Cause part fragment (left) and derived partial FT (right) from second causal step



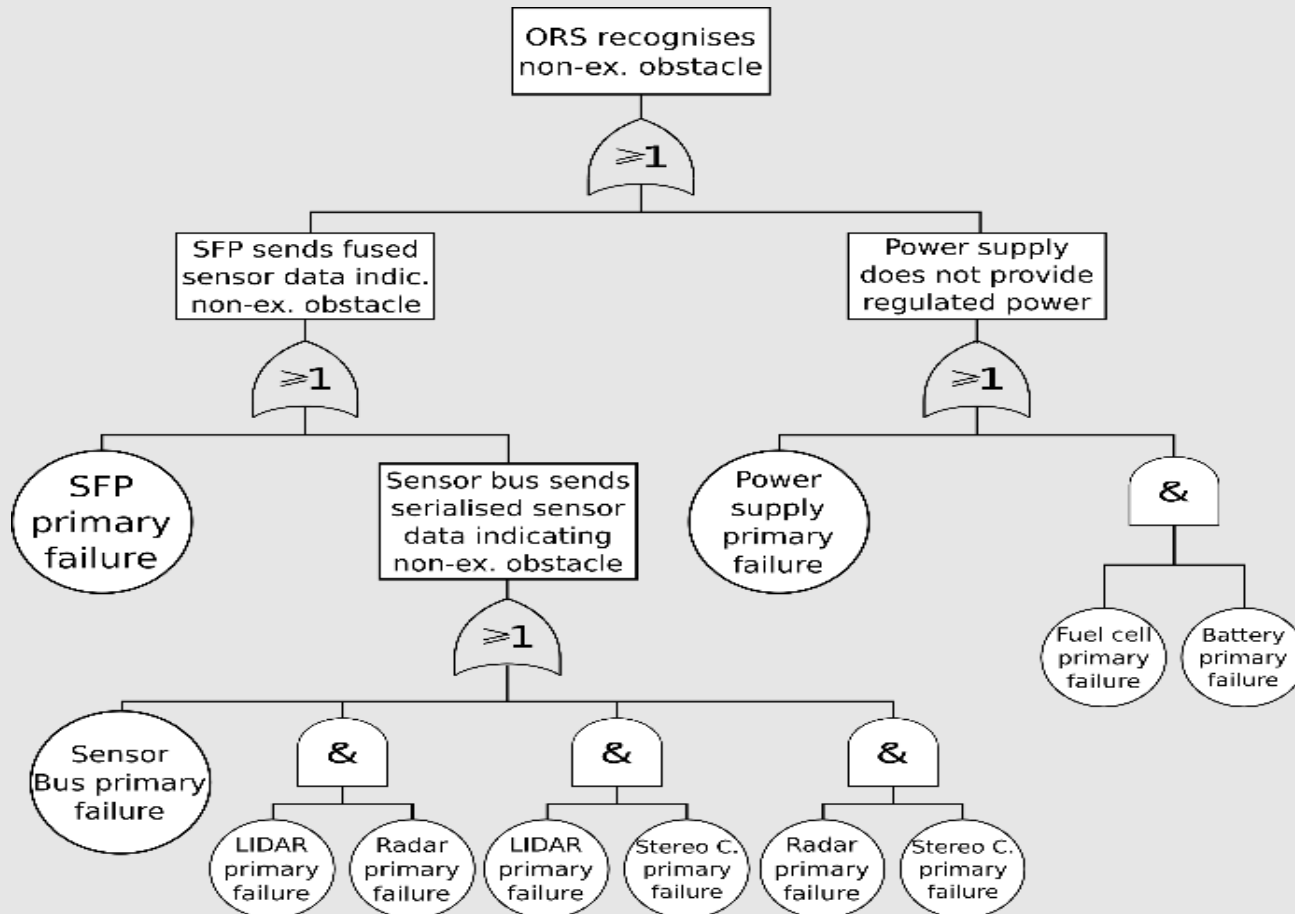
FMC for ORS system function “autonomous emergency braking”



Cause part fragment (left) and derived partial FT (right) from third causal step



Final FT for the example function in failure mode commission



Overview

- > Motivation & Background
- > Extending the Shell Model
- > Short Example
- > Conclusion

Conclusion

- + The Shell Model Analysis has been linked to traditional analysis methods (FMEA, FTA) in a structured methodical approach
- + Structured, methodical way to support safety argumentation as well as documentation
- + The consequence considerations base on classical FMEA
 - > FMEA has been slightly adapted to work with FTA guidewords
 - > FTA guidewords help in finding the error-prone deviations
- + The cause part builds up an FT
 - > Following a backstep-bybackstep approach
 - > Following typical FT design rules (e.g. immediate, necessary and sufficient)
- + Results from these can be further investigated
 - > E.g. (safety-) critical lines from FMEA used as basis for a detailed FMEDA
 - > E.g. (safety-) critical paths/events from FTA used as basis for redesign considerations / hazard mitigation

Conclusion & Prospects

- **Tool support** is extremely necessary, especially for
 - The Shell Model itself
 - Drawing the model (especially for large systems)
 - FMC extraction
 - FMC Consequence Analysis via FMEA
 - Generating and pre-Filling the tables
 - Supporting selection of the critical paths
 - FMC Cause Analysis via FTA
 - Generating the trees
 - Supporting selection of the critical paths and events
 - Supporting via redesign / safety measures

QUESTIONS?

QUESTIONS?