Computer Networks and Communication Systems
Friedrich-Alexander-University Erlangen-Nuremberg
Prof. Dr.-Ing. Reinhard German

Fault Tree Generation from EMF Models

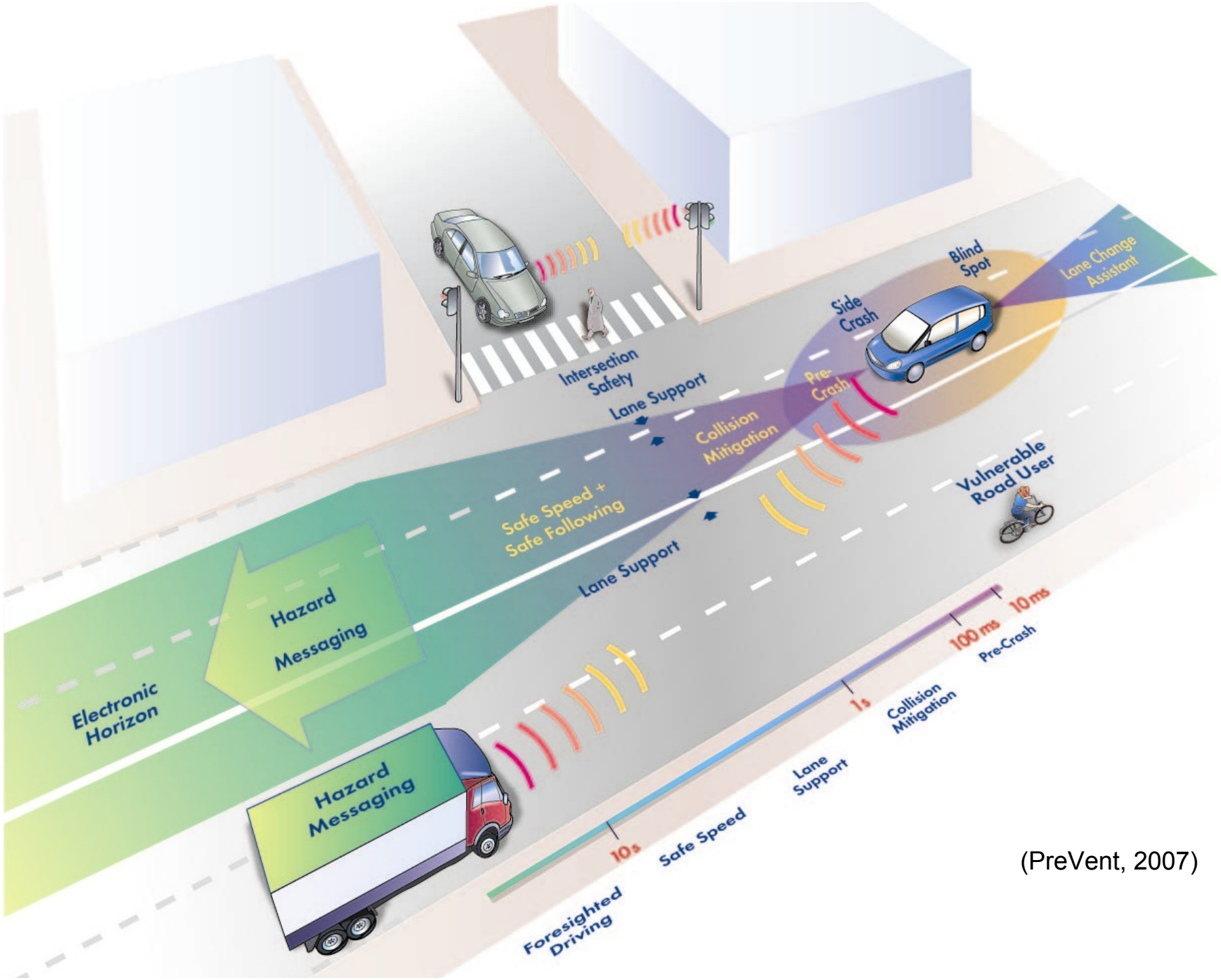
Christoph Lauer[1], Reinhard German[1] and Jens Pollmer[2]

[1]Department of Computer Science 7 – Computer Networks and Communication Systems,
Friedrich-Alexander University, Erlangen-Nuremberg, Germany

[2]Department of Safety Electronics,
Audi AG, Ingolstadt, Germany
{christoph.lauer, german}@informatik.uni-erlangen.de, jens.pollmer@audi.de

Outline

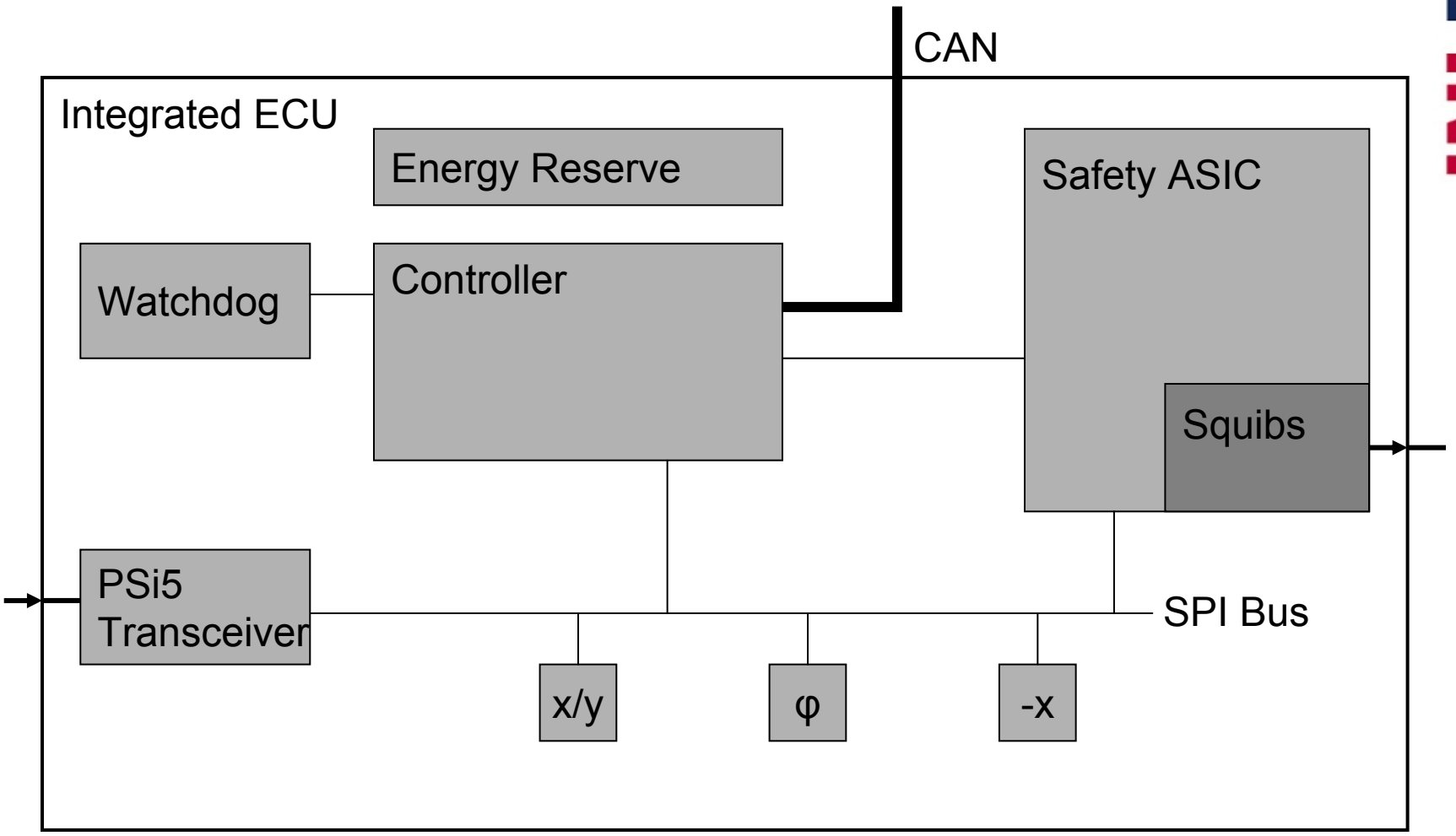
- Introduction
- Integrated Safety Architectures
- System Models
- Fault Tree Generation
- Conclusions and Future Work



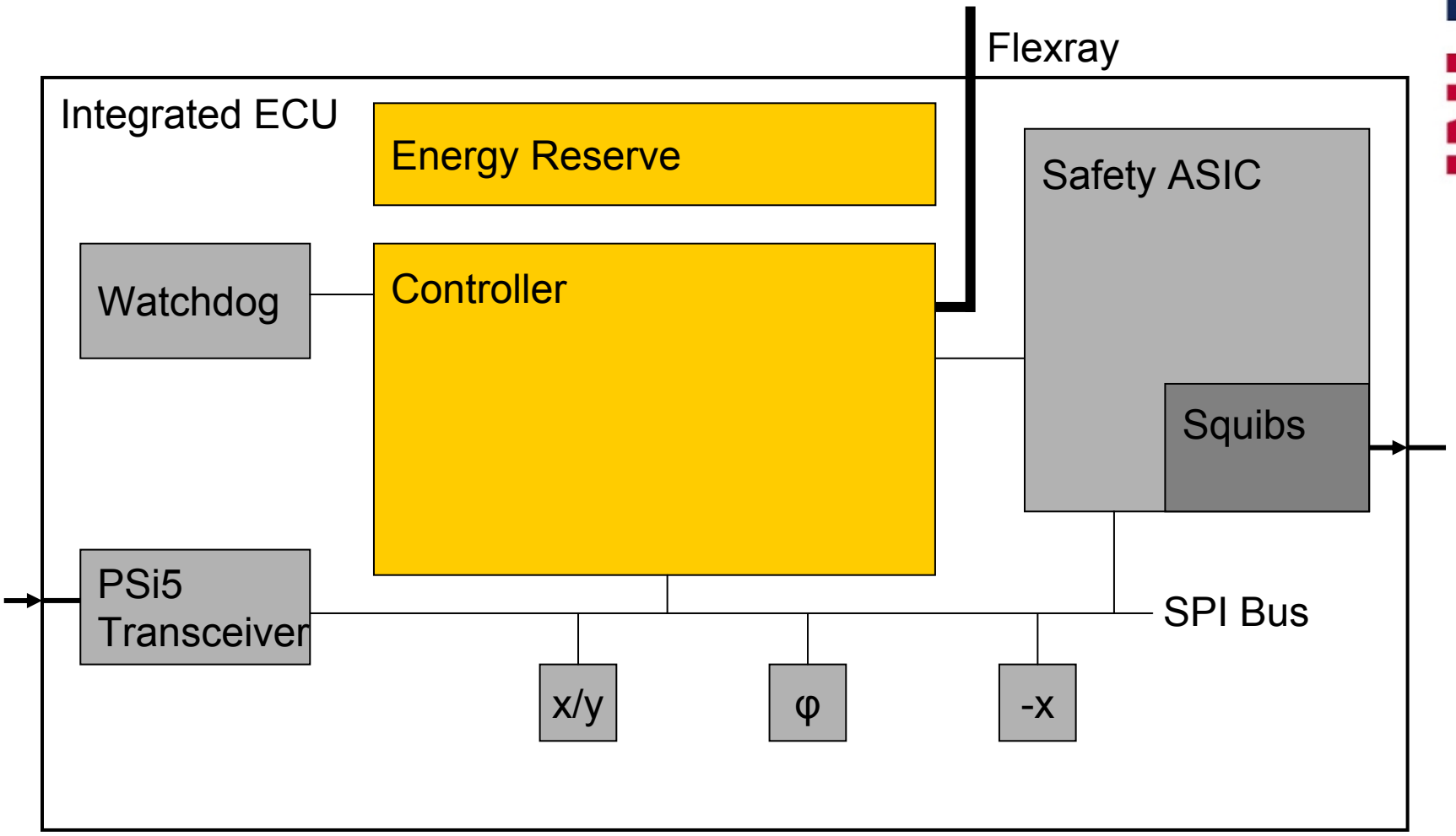
(PreVent, 2007)


PASSAT

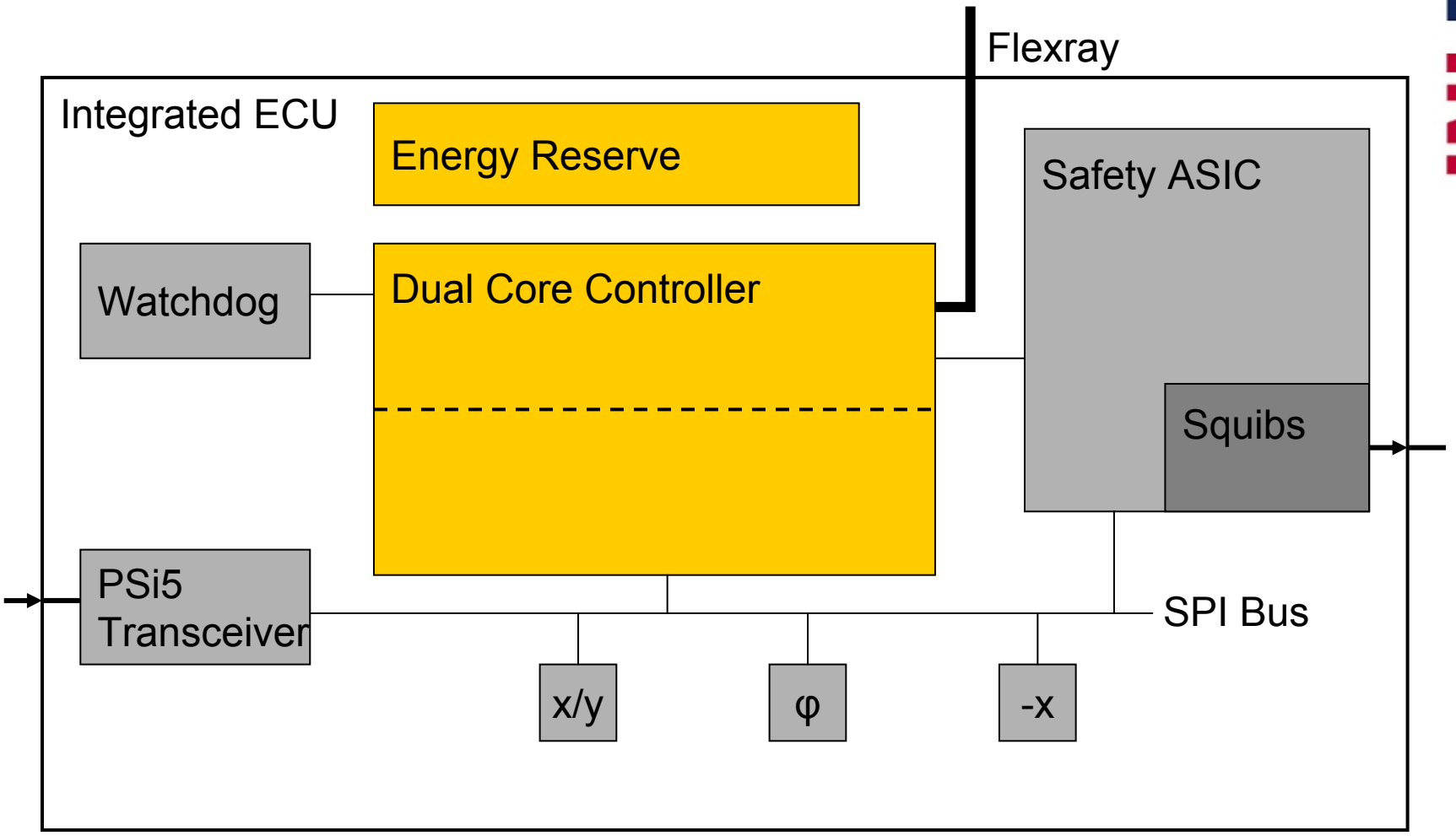
Integrated Safety Architecures in the Automotive Domain (1)



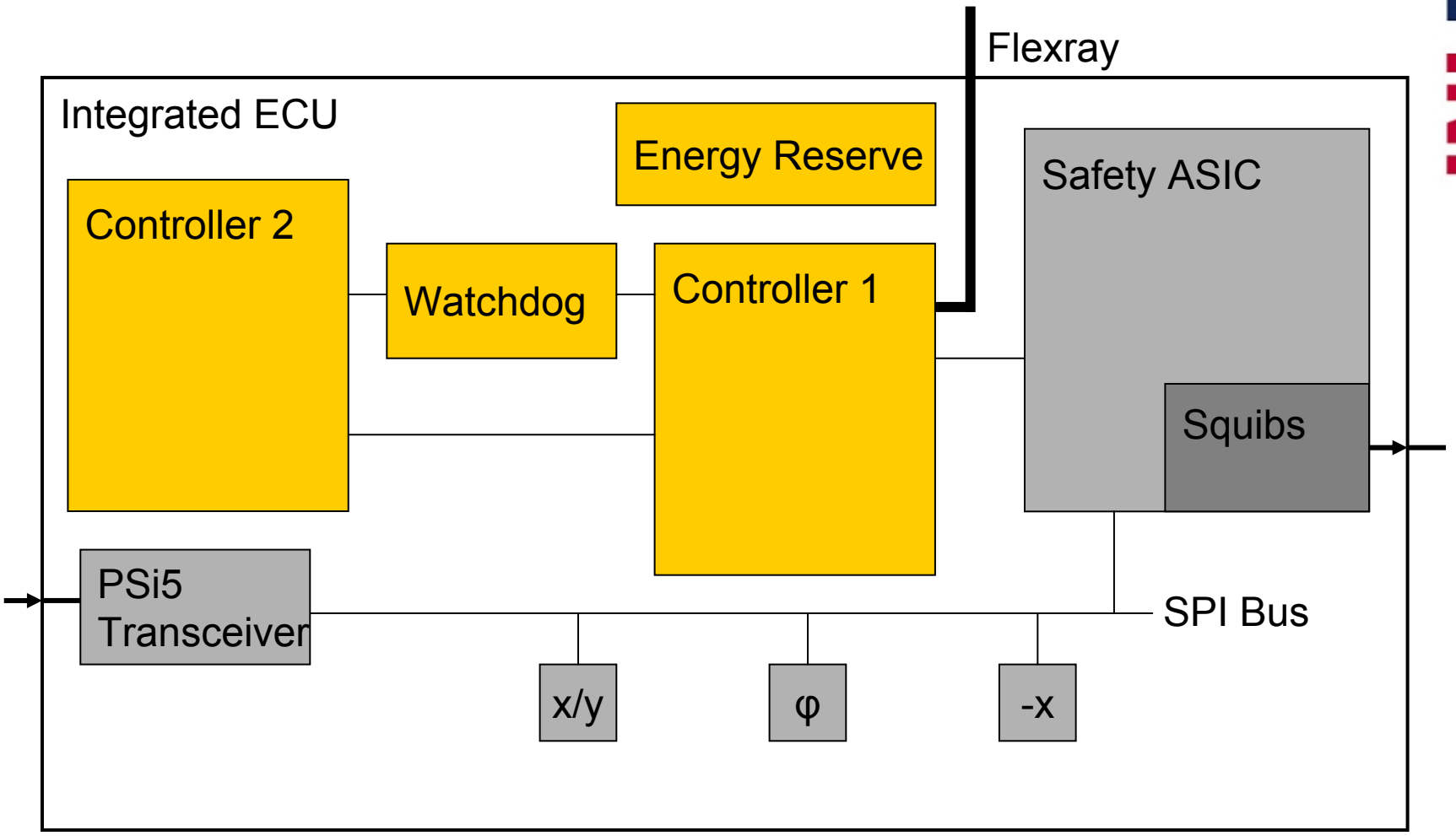
Integrated Safety Architecures in the Automotive Domain (1)



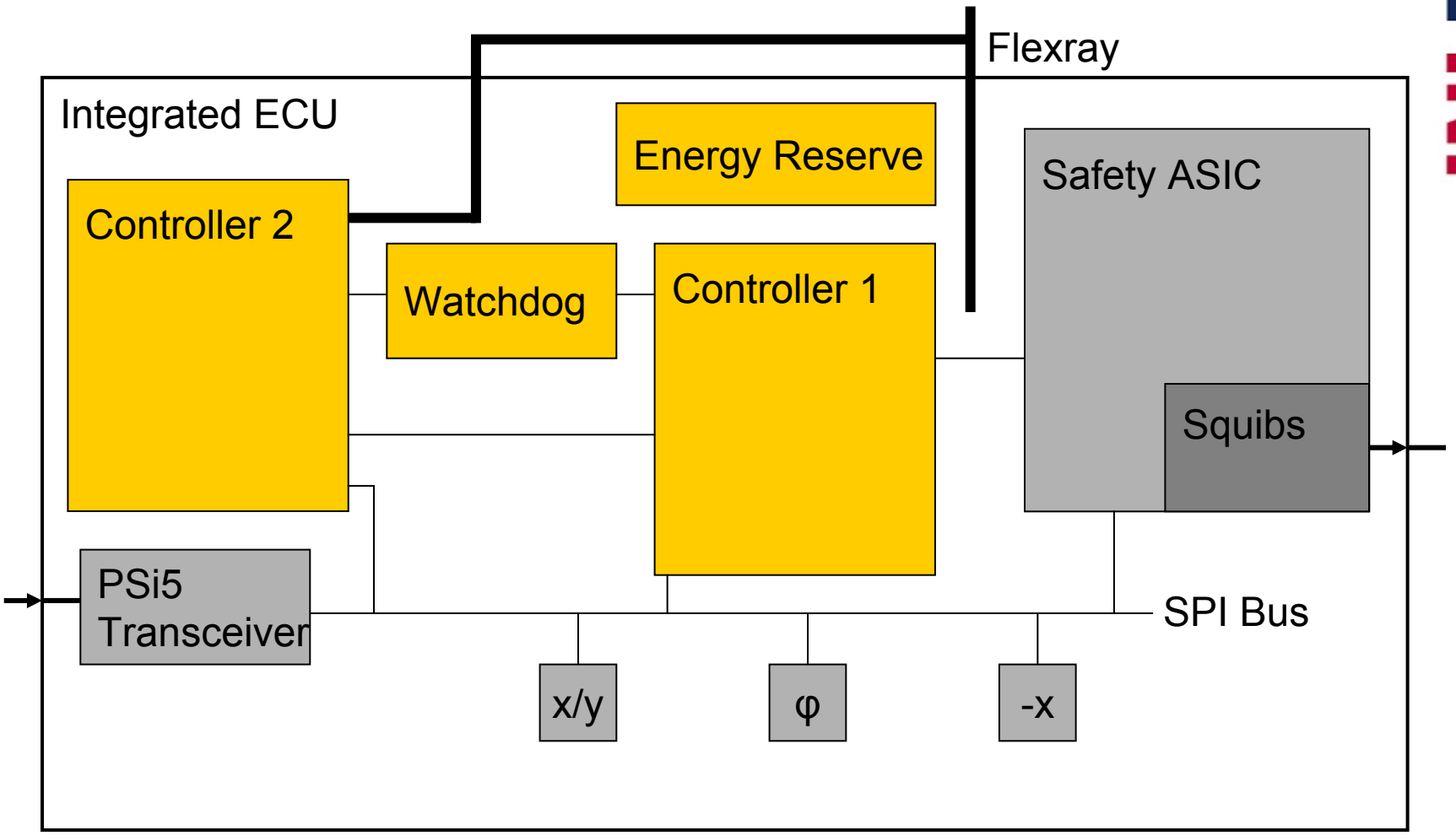
Integrated Safety Architecures in the Automotive Domain (3)



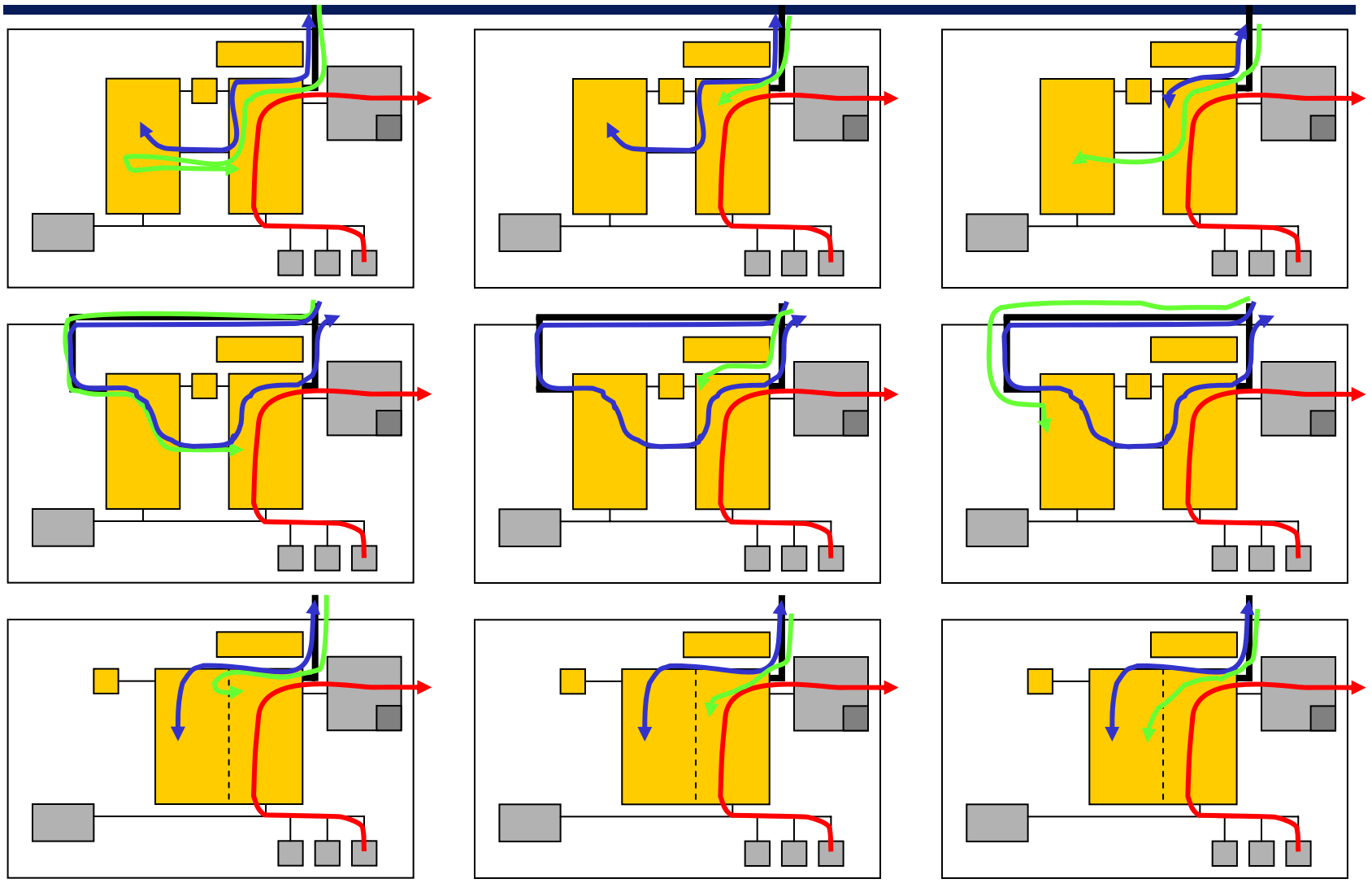
Integrated Safety Architecures in the Automotive Domain (4)



Integrated Safety Architecures in the Automotive Domain (5)



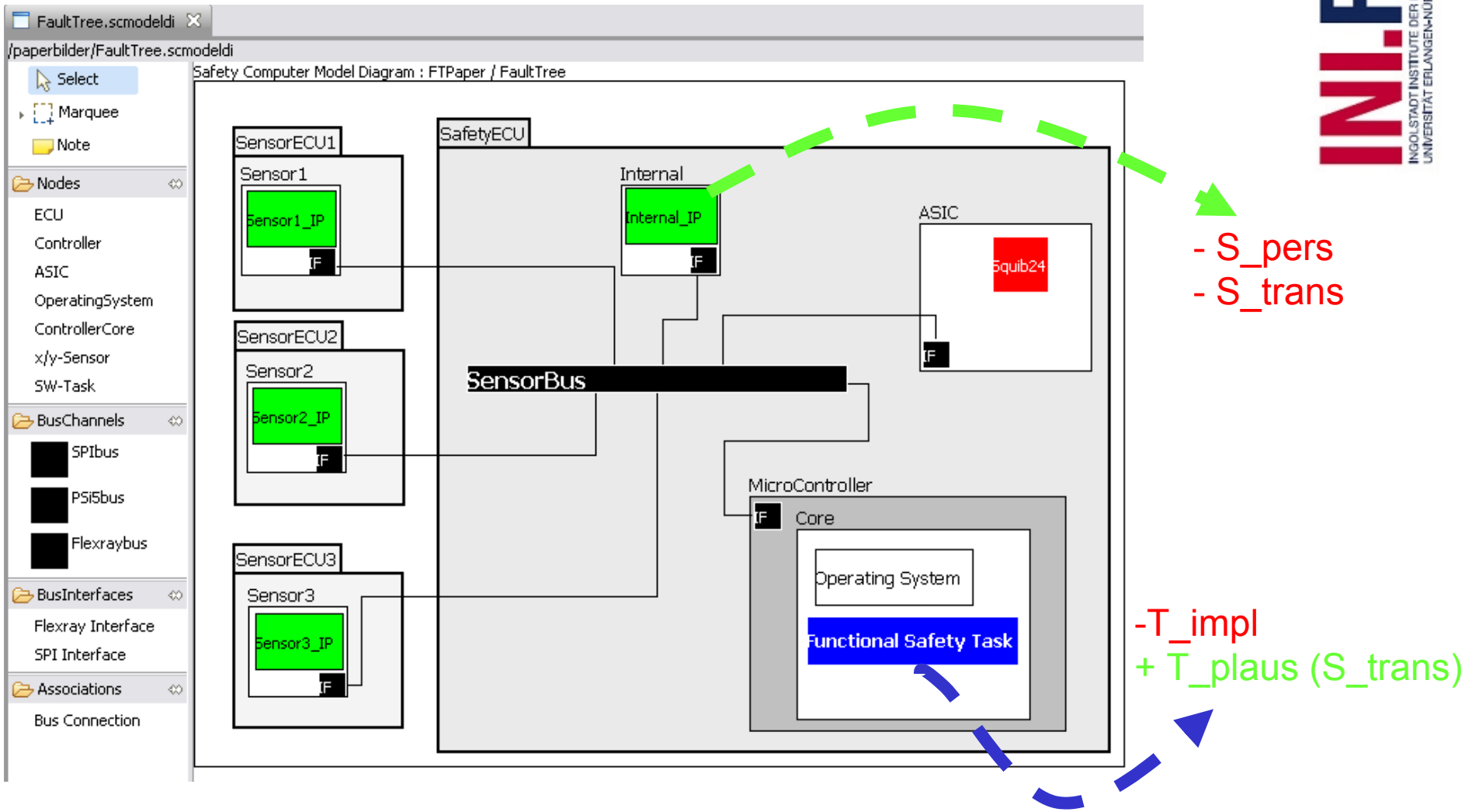
Task Binding Decisions



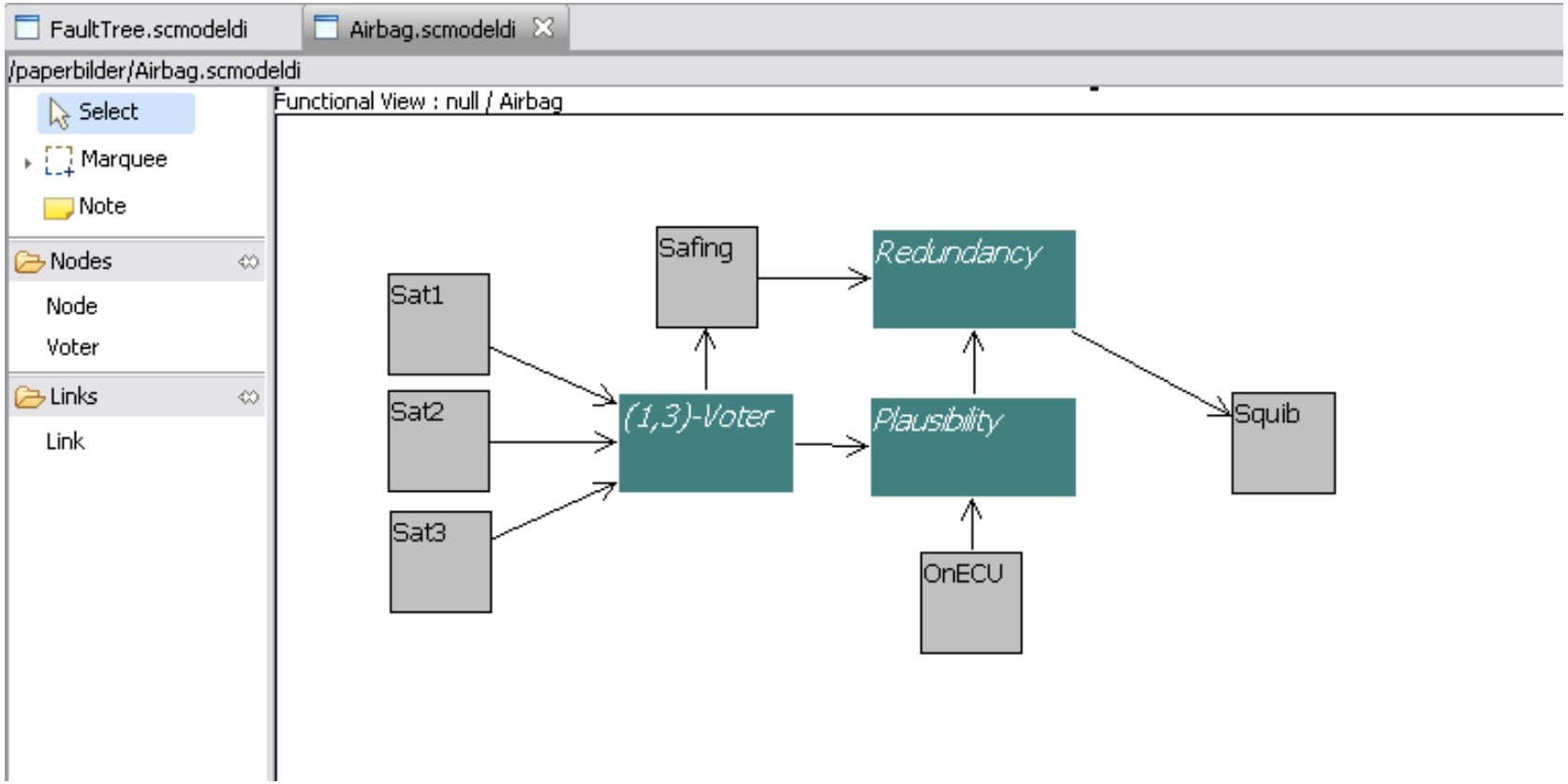
Requirements for Modeling & Generation

- Seperate modeling of system architecture and functional behavior
- Flexible allocation of functional tasks to system nodes
- Automatic generation of fault trees for further analysis using state-of-the-art tools
- No extensive design space exploration

EMF Model Support (1)



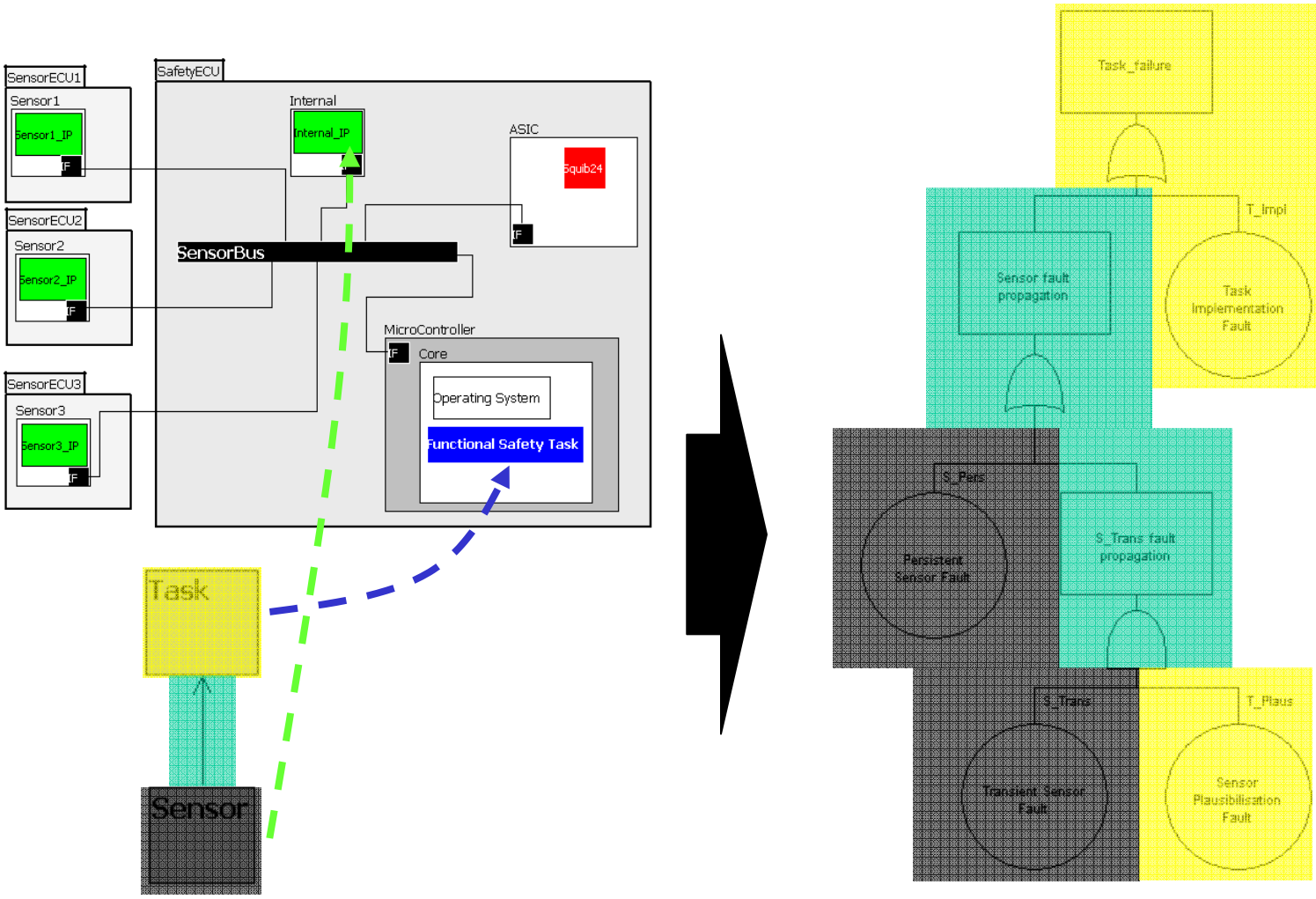
EMF Model Support (2)



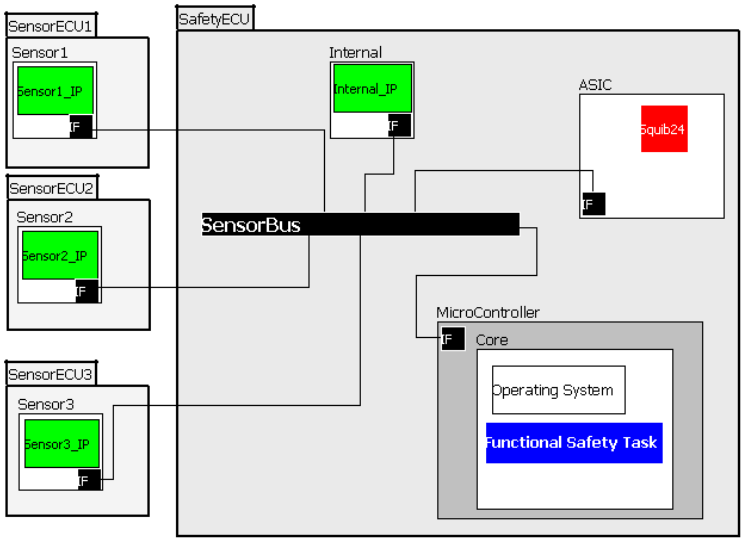
Transformation Rules

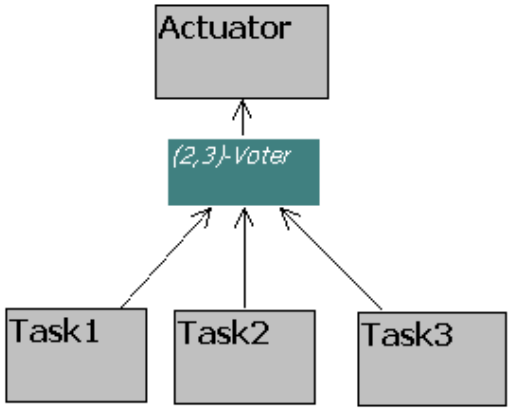
1) Start at top-level event
2) Evaluate top-level event
   a. Get faults from allocated system entity
   b. Add faults of entity directly (via OR gate)
3) Evaluate all incoming edges
4) Evaluate node
   a. Get faults from allocated system entity
   b. Traverse graph to top-level event
   c. Add fault directly (via OR gate) if fault propagates, or add guardian (via AND gate) if fault is not propagated
5) Terminate if no incoming edges exist, else go to 3)

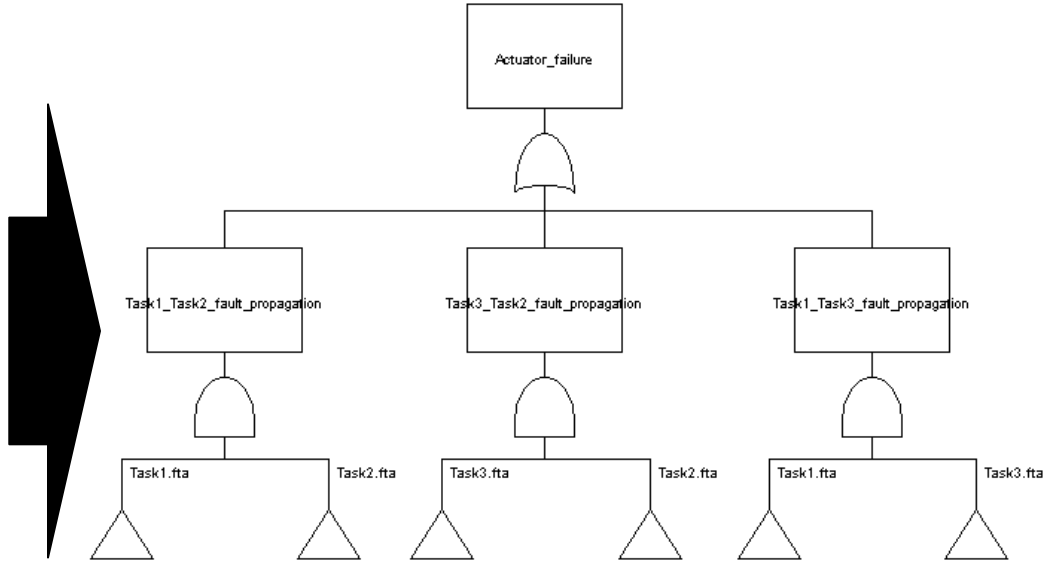
Transformation Example (1)



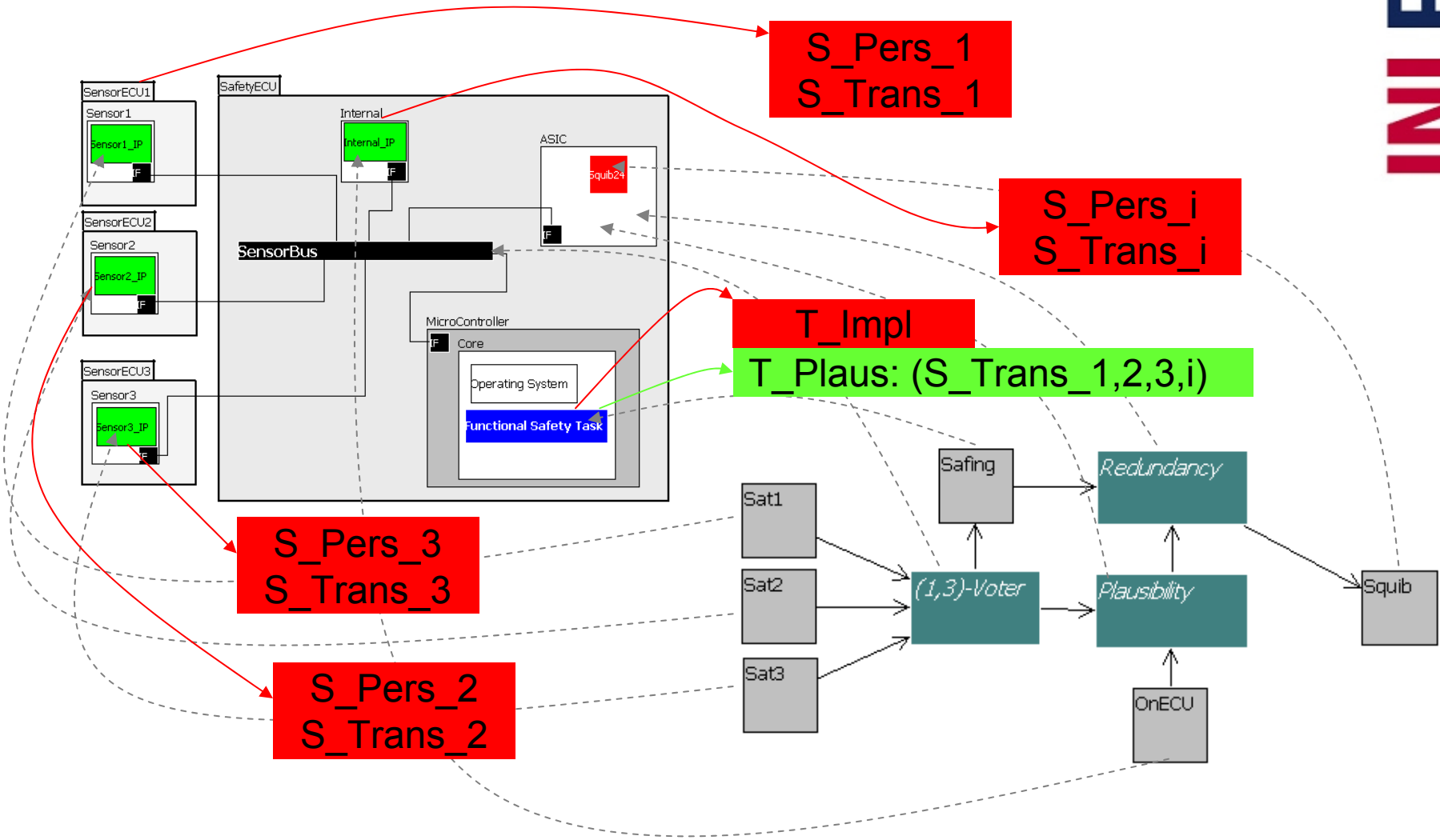
Transformation Example (2)



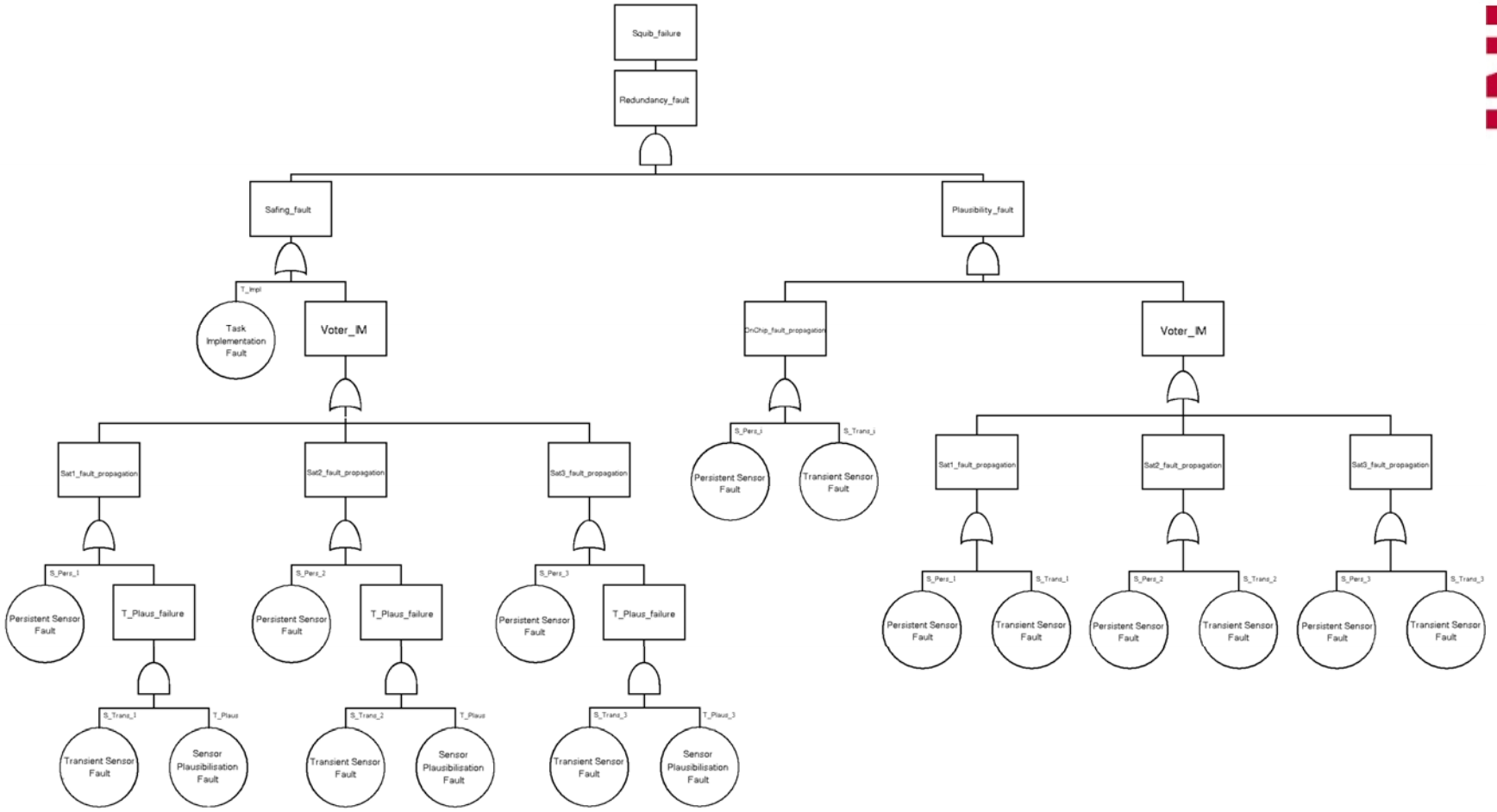




Transformation Example (3)



Transformation Example (4)



Conclusions

- Modeling of system and behavior using the EMF
- Model transformation from separated system model + behavior model to fault trees
- Just a transformation, the algorithm does not „create knowledge“
- Level-of-detail of the fault trees depends on the level-of-detail of the input models
- Method supports analysis of different architecture options at early design stages

Future Work

- Leave the Ecore path for the sake of UML
  - Modeling of the system and the behavior view using MARTE(+ Depandability profile from Bernardi et al. (2008)) or EAST-ADL2
  - Papyrus plug-in for easy modeling without having to cope with UML
- Implementation (!) of interfaces to FaultTree+ (ISOGraph)

Last slide

- Thanks for your attention!