

BA / MA / Prj / WHK

Systemmodell im Embedded System Design

Motivation

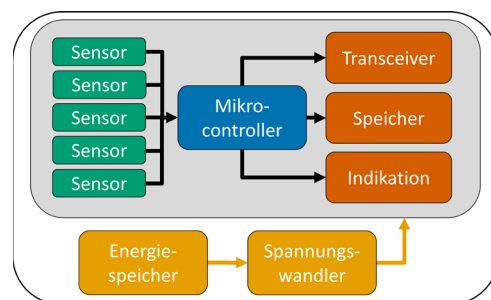
Die stetig steigende Komplexität von eingebetteten mikroelektronischen Systemen, die sich durch eine Vielzahl von Komponenten mit jeweils verschiedenen möglichen Zuständen jeder einzelnen Komponente auszeichnet, führt dazu, dass eine ganzheitliche Optimierung der Systeme nicht mehr, oder nur sehr zeitaufwändig möglich ist. Daher werden heutige Systeme zunehmend modellbasiert entwickelt (Model-Based Systems Engineering (MBSE)). Das Systemmodell stellt dabei die sogenannte „Single Source of Truth“ dar, so dass durch Verweise auf das zugrunde liegende Modell eine konsistente Modellierung gewährleistet wird. Als Modellierungssprache wird typischerweise die Systems Modeling Language (SysML) verwendet. Eine grafische, auf UML 2 basierende, standardisierte Modellierungssprache.

Hier wurde vor kurzem die SysML v2 vorgestellt. Diese wurde nicht aus UML abgeleitet und verfügt, neben der grafischen auch über eine textuelle Notation. Das System kann somit in einem Jupyter Notebook oder in einer Eclipse IDE textuell beschrieben werden. Darüber hinaus steht eine API zur Verfügung, so dass eine Interoperabilität mit anderen Modellen und Werkzeugen gegeben ist.

Ziel und Vorgehen

Es soll daher evaluiert werden, in wie fern ein Embedded System (z.B. ein Sensorknoten) in SysML v2 modelliert werden und wie sich andere Umgebungen (z.B. Simulationstools) andocken lassen. Mögliche Inhalte sind daher:

- Einarbeitung in SysMLv2 (Notation, API)
- Umsetzung eines Systemmodells für z.B. einen drahtlosen Sensorknoten
- Nutzung des Systemmodell als Datenbasis für Simulationsumgebungen
- Zusammenspiel aus Systemmodell und Simulation evaluieren bzw. optimieren



Komponenten eines Drahtlosen Sensorknotens

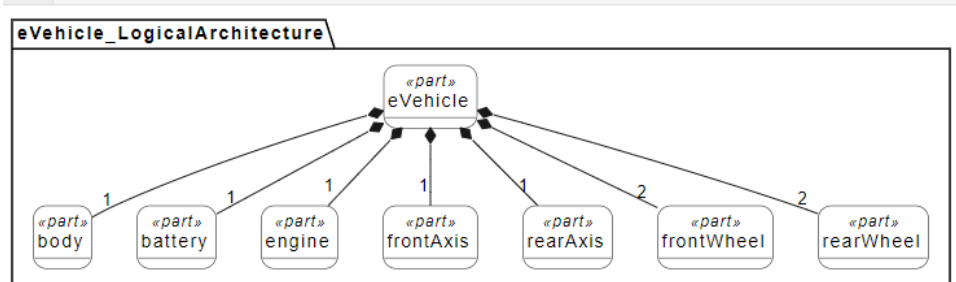
Anforderungen

- Interesse an Modellierung
- Erfahrungen im Embedded System Design von Vorteil

```

1 package eVehicle_LogicalArchitecture {
2   part eVehicle {
3     part body;
4     part battery;
5     part engine;
6     part frontAxis;
7     part rearAxis;
8     part frontWheel[2];
9     part rearWheel[2];
10  }
11 }

```



Beispiel für eine Logic Architecture in SysML v2:

Textuelle Notation im Jupyter Notebook und daraus generierte grafische Darstellung

