



Task Description for Final Thesis (Diplomarbeit or Master thesis)

Degree program: Informatik
Topic: **Automatic interface matching for multi-source programming of embedded controllers**

Make embedded systems resilient to supply chain disruptions

Europe is leading in the field of embedded systems. However, the embedded software running the systems is highly specific with respect to the controller used. Transferring the embedded software from one controller to another or developing the embedded software for two interchangeable controllers in parallel, is a hefty effort as it requires two distinct embedded software projects. To reduce development costs, most embedded systems are designed only for one specific controller (**single source**). Consequently, the producer of the embedded system is solely reliant on the continuous long-term availability and affordability of the controller used. In order to ensure continuous availability, affordability and to boost the resilience through their complete life-span embedded systems must be designed for multiple controllers, what in turn enables an efficient **multi-source** software development with no vendor lock-in.

Want to shake up the embedded systems market?

The heart of a multi-source system is an intelligent process of matching the software interfaces of different semiconductor manufacturers. As a member of the team, your task is to compare the software development kits (SDKs) from two semiconductor manufacturers with each other and to develop a generalizable matching methodology between the software interfaces. As a result, the application code developed on platform A can be ported unchanged to platform B, while ensuring correct execution with low overhead.

Let's get specific

- You will port a given application code, developed on an NXP evaluation board, unchanged to a hardware-compatible Renesas evaluation board
- To do this, you will analyze the SDKs of both manufacturers and extract global content
- You will develop a new layer (source and header) to translate the function calls from NXP to Renesas
- You will automate your manual procedure with a software tool, ideally written in Python
- You will explore approaches based on machine learning for translating between the interfaces (optional)

Referee: Prof. Dr.-Ing. Jeronimo Castrillon
Supervisor: Dr. Steffen Köhler

Prof. Dr.-Ing. Jeronimo Castrillon
(Professor in charge)