Fast Software Performance Evaluation for Embedded Hardware in Component-based Embedded Systems

Michael Pressler[1], Alexander Viehl[1], Oliver Bringmann[1,2], Wolfgang Rosenstiel[1,2]

[1]FZI - Forschungszentrum Informatik
Haid-und-Neu-Straße 10-14
76131 Karlsruhe
pressler@fzi.de
viehl@fzi.de

[2]Universität Tübingen
Sand 13
72076 Tübingen
rosenstiel@informatik.uni-tuebingen.de
bringman@informatik.uni-tuebingen.de

Abstract: We present an approach for the extraction of computational demand of embedded software components and the determination of initial mapping configurations on modern embedded heterogeneous processor architectures. The presented work combines the advantages of component-based design and properties obtained from source-code analysis. The goal is a very fast estimation of execution costs for multiple hardware/software component pairs even before the hardware is physically available.

1 Key Ideas

The growing pressure for shorter time-to-market and development cost reduction of embedded systems issues the need for novel methodologies in embedded system design. In this scope one of the challenges is an efficient system design by composition of software and hardware components including intellectual property (IP) reuse and integration. In this setting of component-based hardware and software system design, an early determination of an appropriate system deployment plays a key role during the system design process. As non-functional execution properties of hardware/software pairs strongly depend on their implementation and architectural characteristics, a fast estimation of execution costs for hardware/software component pairs could tremendously speed-up an early exploration process.

Existing source-code of embedded software systems enables a white-box bottom-up analysis of execution properties while component-based approaches help designers in reusing software components and separate them from low-level hardware-dependent software. The challenge tackled by our approach is to bring together both worlds and combine their strengths and advantages. The elaborated approach and framework allows the determination of initial mapping configurations for embedded software components on modern embedded processor architectures integrated into heterogeneous platforms.

Our approach aims to extract and quantify ***hardware-independent computational demand (HIC)*** from software source code and define a transition to gain ***hardware-specific execution costs (HSE)***. One main characteristic of our approach is that we extract computational demand for each software component from the source code. This has to be done only once. There is no need for target compilers or binary tools. We execute the application on the development platform to obtain data dependent but hardware-independent execution characteristics of the application. If the developer changes components of the hardware platform or their configuration, only the transition to the HSE needs to be recalculated. The basic approach of the analysis is depicted in Figure 1.

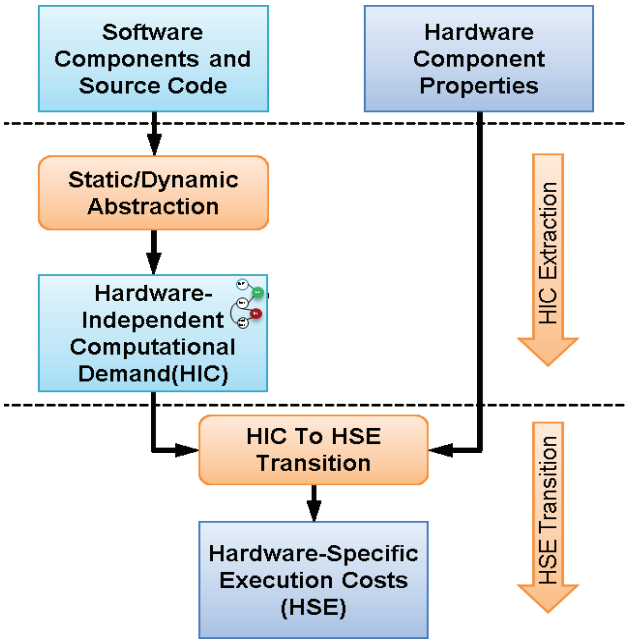

Figure 1: Key Ideas

The HSE determination is a transition based on abstract hardware specifications. The transition calculates the HSE for each possible hardware/software pair. This estimation requires only an abstract hardware specification that can be extracted from the data sheets of the hardware platforms. This means much less effort than implementing virtual models or using target compilers and binary tools for worst-case execution time analyzers. The transition only needs seconds which makes the approach much faster than the determination of HSE via virtual prototypes or ISS and can speed up the design process in complex software systems on heterogeneous hardware components before the initial mapping configuration is available.

The HSE values are used for an initial mapping determination approach. Experimental results show that the estimation is very fast and accurate enough to help designers making initial system configuration decisions.