Extending Code Generation to Support Platform-Independent Event-B Models

Asieh Salehi, Michael Butler and Colin Snook
University of Southampton

6th Rodin User and Developer Workshop
May 23th, 2016, Linz, Austria

Summary:
Code generation was introduced in the Event-B formalism to address the gap between the lowest level Event-B refinement and an implementation. However, the code generation supports generating a single implementation for a refined Event-B model. This results in dependency between the Event-B model and target platform architecture. To address this limitation, we present an extension of the Event-B code generation technique supporting generation of different platform-specific implementations from the same Event-B model. A refined Event-B model is treated as platform-independent through parameterisation. The platform parameters are instantiated in order to generate a platform-specific implementation and these are used by the code generator to produce an implementation that is tailored to the platform.

We applied our approach to model an embedded Run-Time Management (RTM) system; and generated three different RTM implementations for three hardware platforms with different specifications.

Motivation:
The figure presents generation of part of the RTM implementation for two ARM hardware platforms: Cortex_A8 and Cortex_A7, from a platform-independent Event-B action. The update_qTable event updates the look-up table used in the machine learning algorithm during run-time. The number of qTable columns depends on the number of frequencies each platform supports. The platforms support different number of frequencies (N); The Cortex_A8 supports 4 frequencies, whereas Cortex_A7 supports 13 (frequency values specified as constants $FREQ_i$). There is a guard of the update_qTable event indicated as an expanding guard. Variable $N$, used in the expanding guard, is instantiated during code generation and results in generating a collection of $N$ conditional branches in C to modify the qTable with $N$ columns.

References:
