PSMGen: Automatic generation of power state machines

Alessandro Danese [(1)], Daniel Lorenz [(2)], Graziano Pravadelli [(1, 3)]

[(1)] Department of Computer Science University of Verona, Italy – {name.surname}@univr.it
[(2)] OFFIS Institute for Information Technology, Germany – lorenz@offis.de
[(3)] EDALab s.r.l., Italy – graziano.paravadelli@edalab.it

Introduction

Power State Machines (PSMs) are a well-known approach to model and simulate the time-based energy consumption of IP cores for early virtual prototyping of system-on-chips. However, despite of the wide adoption of PSMs, in the most of the works either the presence of PSMs is assumed or they are manually defined starting from a more or less precise knowledge of the functional blocks composing the target IP [1].

To allow a more precise definition of PSMs, we present PSMGen [2], a tool that implements a fully-automatic methodology for PSMs' generation and an efficient statistical approach for their simulation.

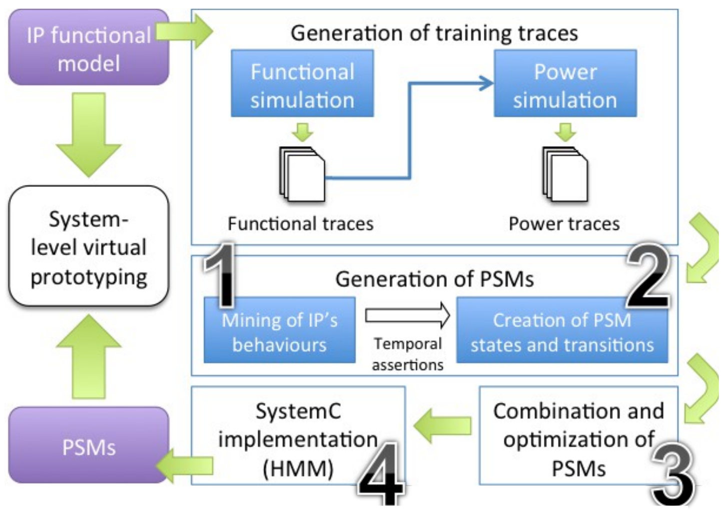

Figure 1: PSMGen overview

PSMGen overview

The assumption under the use of a PSM to model the dynamic power of an IP is that there is a correspondence between a specific functional behavior (characterized by a switching activity) and its energy consumption. Based on this assumption, the execution flow of PSMGen consists of four main steps (Fig. 1):

1. *Mining of the IP's functional behaviors:* Given a set of functional traces, the behaviors of the corresponding IP is initially captured through a set of temporal assertions that are automatically generated by a mining procedure [3].
2. *Generation of PSM states and transitions:* A PSM is generated for each functional trace by associating the power consumption to each of the behaviors extracted at step 1. The power consumption of each state is characterized according to a calibration procedure that exploits a set of reference power traces.
3. *Combination and optimization:* The PSMs generated for each functional trace at step 2 are independent from each other. Moreover, they are in the form of chains of states, where each state has a unique successor and a unique predecessor. A reduced and optimized set of non-deterministic PSMs is then obtained by merging PSMs' states according to their similarity from the point of view of the energetic behavior.
4. *Simulation of PSMs:* Finally, a SystemC model of the PSMs generated at step 3 is automatically synthesized to allow the system-level power simulation of the target IP. The simulation is based on a statistical approach, exploiting Hidden Markov Models (HMM), which is applied to efficiently manage not-deterministic choices during the PSMs traversal.

PSMGen does not require to instrument the functional model of the target component and it can be applied even in case of black-box IPs.

References

[1] D. Lorenz, K. Gruettner, and W. Nebel, "Data-and state-dependent power characterisation and simulation of black-box RTL IP components at system level," Proc. of IEEE *DSD*, 201
[2] A. Danese, I. Zandonà, and G. Pravadelli, "Automatic generation of power state machines through dynamic mining of temporal assertions" Proc. of ACM/IEEE DATE 2016.
[3] A. Danese, T. Ghasempouri, and G. Pravadelli, "Automatic extraction of assertions from execution traces of behavioural models," Proc. of ACM/IEEE DATE 2015.