AIPHS

***A**dapt**I**ve **P**rofiling **H**ardware **S**ub-system*

G. Valente, A. Bufalino, V. Muttillo, L. Pomante, M. Faccio, F. Federici: Center of Excellence DEWS, Università Degli Studi Dell'Aquila, ITALY

Contacts: (giacomo.valente@graduate.univaq.it, andrea.bufalino@student.univaq.it, vittoriano.muttillo@graduate.univaq.it, luigi.pomante@univaq.it, marco.faccio@univaq.it, fabio.federici@univaq.it)

Introduction: Monitoring a system during its life cycle can provide useful information to trace a profile of behaviors at different level of granularity. Results of monitoring action are data that can be used to evaluate some metrics, in order to give insight of the system behavior. This operation can be the starting point to drive different actions, such as the reconfiguration of the platform or a change in the scheduling policy. In the area of *Embedded SoC*, where platforms are composed by heterogeneous elements opportunely interconnected, automatize system observation leads to the requirement that the monitoring system should be customizable, since different behaviors can be required to be analyzed. In this context, run-time monitoring systems on reconfigurable logic offer an interesting perspective to look for this possibility of customization.

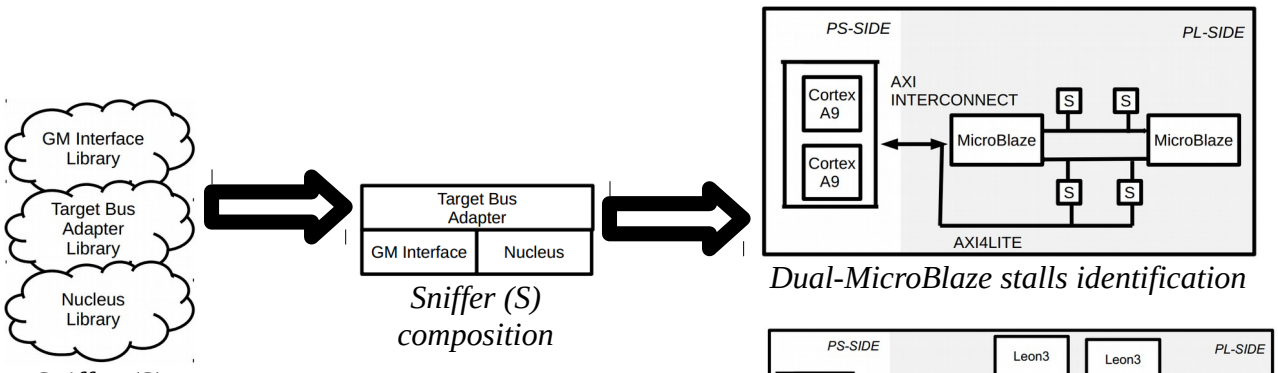
Objective: The main goal of this demo is to present a smart monitoring system, called *AIPHS*, that can be composed starting from a library of elements that allow to customize the system to observe different behaviors.

Demo Description: This demo presents *AIPHS* by showing:

- a customization for stalls identification in a message passing scenario (based on multiple MicroBlaze that executes a bare-metal *FFT* application)

- a customization for bus utilization monitoring in a symmetric multi-processing system scenario (based on four Leon3 running a custom Linux kernel).

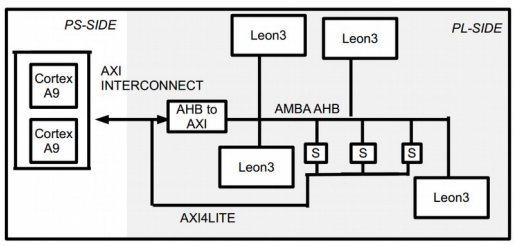
The whole development flow (and related prototypal *EDA* tools), that starts exploiting a library of elements to compose the desired hardware profiler and that leads to the introduction of such a profiler in the target architecture to allow profiling data collection and analysis will be shown. Moreover, a comparison among different functionalities will be illustrated. Both systems will be illustrated by using *Zynq7000 SoC*.



Sniffer (S) Library

Sniffer (S) composition

Dual-MicroBlaze stalls identification



Quad-Leon3 performance monitoring

This work has been partially supported by the Artemis-JU ASP 2011 CRAFTERS (GA 295371) and Thales Italy.