Evolving Computer Programs for Processor Core Power Estimation

Garrison W. Greenwood	Xiaobo Hu	David B. Fogel
ECE Dept., Portland St. Univ.	CSE Dept., Univ. of Notre Dame	Natural Selection Inc.
Portland, OR	Notre Dame, IN	La Jolla, CA
(503) 725-3806	(212) 631-6015	(858) 455-6449
greenwd@ece.pdx.edu	shu@cse.nd.edu	dfogel@natural-selection.com

Problem Statement: Pre-designed intellectual property cores provide high performance with a greatly reduced design cycle time. A good example are the processor cores, such as the Advanced RISC Machine (ARM) core, which are now extensively used in personal digital assistants and hand-held games.

Processor cores are now commonplace, which means end users have to conduct design trade-offs to make the best possible choice. Power consumption may be a critical factor in making that choice, but most core vendors provide little assistance in making power estimates because power consumption is application dependant, and accuracy requires detailed information on the target application. Unfortunately, time-to-market considerations or intellectual property concerns may make this information difficult or even impossible to get. Indeed, in hardware/software codesign environments the very definition of application software is still evolving. Hence, our problem of interest is stated by the following question:

> *Is there some way of reliably estimating the power consumption in a processor core that takes into account the customer's application—even if detailed application information is missing?*

Objective: Our objective is to develop a design automation tool that answers the above question in the affirmative. It will work with both hard and soft cores and is intended to be part of a design automation tool furnished by the core provider along with the functional tests and other support documentation. This results in a complete and effective test methodology for the customer. Finally, our tool will be particularly useful to designers trying to resolve hardware/software partitioning problems.

Approach: Our approach is simple but effective: the customer is asked to specify an upper limit on the power dissipation and a maximum program size (e.g., assembly language instruction count or lines of C code). If the core has any functional units, such as a multiply-accumulator, then anticipated usage expressed as a percentage will also be needed. Any program meeting the size and usage constraints is called a *feasible program*. The customer will also be asked to input the processor clock speed.

We will use *genetic programming* (GP)—i.e., an algorithm that attempts to emulate Darwinian principles found in nature—to evolve assembly language programs. The GP is designed to specifically render only feasible programs. Otherwise, the evolved programs are completely random.

Each evolved program is used as input stimuli to a power estimation software[1] to obtain the power consumption value for the program. The goal is to evolve a feasible computer program that exceeds the customer's stated power budget. If we can successfully evolve a program that exceeds the input power upper limit, then the customer knows the processor core under consideration could exceed the power budget in the target application. If we are unsuccessful, the value we do obtain becomes a least upper bound on power dissipation. Either way, the customer has reliable power consumption estimates—tailored to their target application—that can be invaluable when making a choice between several different processors.

Depending on whether the core is a hard core or a soft core, different power estimators may be used. Furthermore, power estimators based on models at different levels of abstraction vary significantly in terms of accuracy and efficiency. Through our research, we will investigate the impact of such tradeoffs in the evolution process and propose an effective combination of the different power estimators.

Novelty: Our approach estimates hard or soft core power dissipation in ways that consider a customer's target application, while requiring minimal customer input. Moreover, it can be easily incorporated into commercially available design automation tools. This capability doesn't exist in any current design automation system.

Research Output: We intend to use an ARM processor soft core in a real-world embedded application as a test case. The GP source code (written in C++) and all test results will be made freely available to SRC members over the Internet.

Financial & Labor Estimates: The annual budget is approximately $40,000. The project team will also require one graduate student.

[1] We will use the Synopsys Inc. power estimation tool suite. These Synopsys tools have integrated power models.