PowerMixer^{IP}: Power Analysis Framework for SoC Designs

Chia-Chien Weng[1] Shan-Chien Fang[2] Chen-Wei Hsu[1] Jia-Lu Liao[1] Shi-Yu Huang[1]
[1]Dept. of Electrical Engineering, National Tsing Hua University, Hsinchu, Taiwan
[2]TinnoTek Inc, Hsinchu, Taiwan (service@tinnotek.com.tw)

INTRODUCTION

System-on-a-Chip (SoC) designs consist of heterogeneous design components such as processors, DSPs, memories, bus, bridges, and other IPs with various functions as well as software components. The SoC power consumption will continue to grow rapidly for the next decade due to the ever-increasing size and complexity in a SoC. In order to avoid the problems induced by high power dissipation, and to achieve power reduction, a power estimation tool is needed for analyzing the power consumption of SoC designs during the design process.

As shown in Fig. 1, we have developed four power analysis tools: (1) *PowerBrick*, a power characterization tool to construct power libraries for standard cell library and memory compiler, (2) *PowerMixer*, an RTL/gate-level power estimator for large logic design, (3) *PowerMixer*^{IP}, an IP-based power model builder to build power models for general IPs as well as processor IPs, and (4) *PowerDepot*, an ESL power estimation tool to enable super-fast system-level SoC power estimation. Equipped with these highly automatic tools, one is able to drastically reduce the effort and time spent in building the power analysis environment for SoC designs. The simulation speedup can be up to 2,400X comparing with traditional simulation methodology, while retaining very high accuracy.

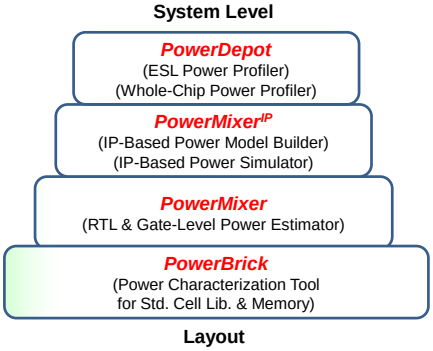

Fig. 1: Our power analysis framework.

PowerMixer^{IP}: IP-based Power Analysis

PowerMixer^{IP} can build an IP-based power model for a given IP design, and integrate this IP-based power model easily into a SoC platform to perform IP-based power analysis. In the IP-based power modeling process, *PowerMixer*^{IP} can build the power models for various kinds of IPs such as processor cores, SRAM, bus, bridges, EMDMA, and so on. As shown in Fig. 2, *PowerMixer*^{IP} first invokes *PowerMixer* to perform gate-level or RTL power simulation to produce accurate power waveform of the IP under modeling with some user-specified training VCD files, and then builds operation-mode-based power model for a general IP or instruction-level or even pipeline-stage-accurate model for a processor. The input files of *PowerMixer*^{IP} include the training VCD files for charactering the power models, and some configuration files, i.e., operational-mode file (.OMF) and key signal file, used to describe the operation modes, such as active mode, idle mode when modeling general IPs, or to indicate some key parameters of the processor, such as pipeline depth, instructions per fetch, instruction type registers, and program counter register when modeling processors.

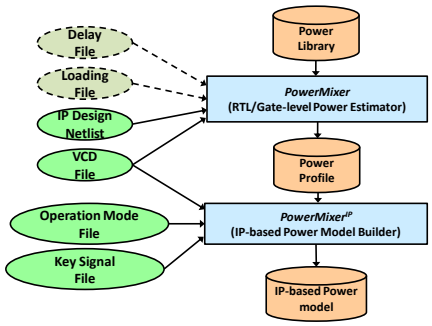

Fig. 2: IP-based power modeling flow.

Once the power models of various IP designs have been built, the user can invoke *PowerMixer*^{IP} again to run IP-based power simulation for the SoC platform. During this process, the inputs files include the gate-level SoC netlist, the power models files (.PMF), the VCD file, and the standard cell power library. The VCD file can be essentially dumped, which means only the key signals of the IP need to be recorded in the VCD file. Hence, the VCD file size can be greatly deduced so as to speed up the simulation process. The standard cell power library is used to calculate the power consumption of the logic cells which has not been built as a power model such as glue logic.

PowerDepot: ESL Power Analysis

The power models generated by *PowerMixer*^{IP} can be integrated into the ESL platform. Then, by re-running the ESL simulation, the power profile of the ESL system can be produced. As shown in Fig. 3, there are three steps in this process: (1) Power monitor generation, (2) Power monitor insertion, and (3) ESL re-simulation. The power monitor is implemented as a SystemC module used to monitor the power consumption of the IP module in the ESL platform. The *PowerDepot* first reads the power model files produced by *PowerMixer*^{IP} and the ESL-to-HW name-mapping file, which indicates the mapping of key signal names and module names between ESL and gate-level/RTL, to generate the power monitors automatically. Then, each power monitor is inserted into the SystemC module of each corresponding IP in the ESL platform. Finally, the SystemC-based ESL platform is re-compiled and re-simulated to produce the power profile of each IP as well as the power accumulation of the whole ESL system.

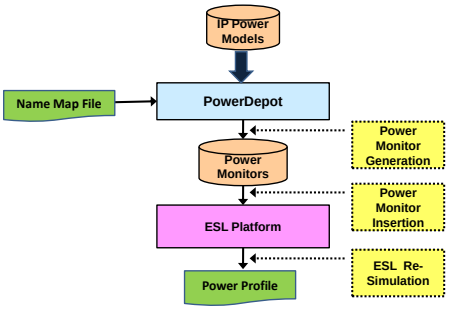

Fig. 3: ESL power analysis flow.