

Building Various Levels of SOC Architecture Exploration Environments: System Level Simulator, Emulator and FPGA Prototype Board

Gi-Ho Park, Chang-Hoon Oh, Jong Wook Kwak , Hyun-Min Kyung, Jung-Bin Im, Sung Yong Cho, WooKyeong Jeong, Tae-Jin Kim, Sung-Bae Park

Processor Architecture Lab, SOC R & D Center, System LSI

Division, Semiconductor Business, Samsung Electronics, Yongin-City, Kyeonggi-Do, Korea

giho.park, ch.oh, jongwook.kwak, hyunmin.kyung, bin5000.Im, sy77.cho, wk.jeong, taejinkim,

sung.park@samsung.com

1 Requirements for Various SOC Architecture Exploration Environments

As the physical gate-count in System-On-Chip (SOC) system increases and system design complexity grows rapidly, performance verification in the early design stage is very crucial in designing complex SOC chips. For the conventional SOC design, SOC architects usually hand-calculate the memory traffics of a specific function IP, such as a H.264 decoder, based on the IP specification and the algorithm analysis to design the bus system architecture. For the complex SOC design with multiple cores, especially the programmable core performs large applications, such as multimedia codec processing, it is very essential to understand the dynamic behaviors of the system such as the bus traffic and latency. As the advanced on-chip bus system like ARM AMBA3 AXI [1] bus based architecture and sophisticated memory controllers are used to design complex SOCs, it becomes more difficult to apply hand-calculation based architecture design approach for the current SOC design.

System level simulation tools such as ARM RealView SOC Designer [2] and CoWare Platform Architect [3] have been provided for the system architects to rapidly explore their SOC architecture with the capability for the detailed analysis and evaluation against the required specification. These tools usually have features to animate the transactions between various components in the SOC to yield a high level view of traffic throughout the system and to set breakpoints on almost any part of system, including different interconnect points, memory and register locations as well as on assembly or source code lines. However, the speed and accuracy of these system level simulation tools are not sufficient for designing very complex SOC performing multimedia processing which requires very long simulation cycles to measure accurate performance. So, it is very difficult to design the SOC architecture by using only these system level simulators. To overcome these drawbacks of the system level simulator, we are building various levels of architecture exploration environments to verify the performance of the target SOC design with the high accuracy and efficiency. This paper presents these architecture exploration environments and considerations for building and utilizing these environments.

2 Various SOC Architecture Design Environments

2.1 Target Platform SOC Architecture

We are developing a platform SOC for the mobile multimedia processing. The platform SOC named SAVm IV has a heterogeneous multi-processor architecture with an ARM1176 processor and a StarCore SC2400 DSP processor. The StarCore DSP processor mainly performs multimedia processing such as the video/audio processing. Figure 1 shows the architecture and IP features of the SAVm IV SOC platform.

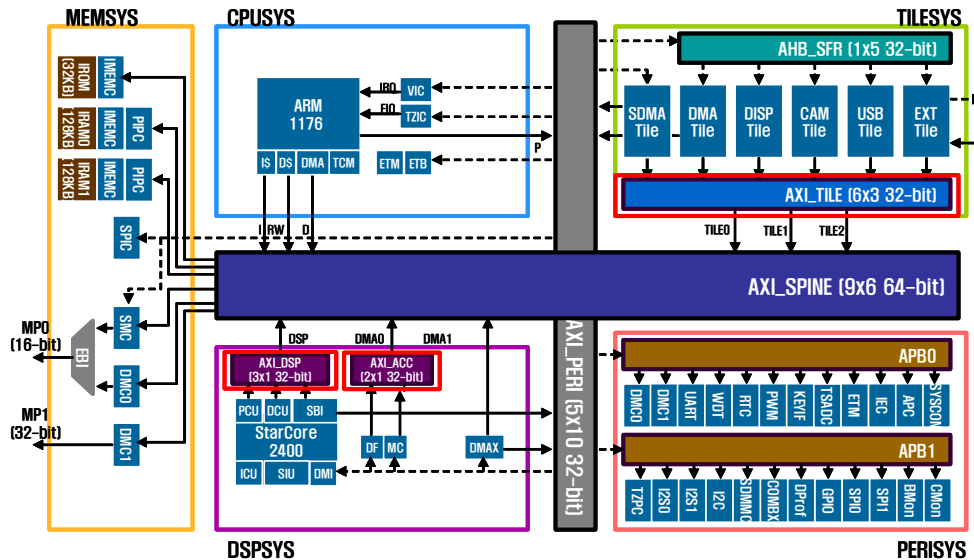


Figure 1. SAVm IV SOC platform

2.2 Various Environments for SOC Architecture Design

We are building various levels of architecture exploration environments as shown in Figure 2. Those include a hardware emulation environment based on the Cadence Palladium hardware emulator and an FPGA prototype board with a performance analysis unit (PAU) as well as a system level simulator.

We built a system level simulation environment based on the ARM RealView SOC Designer tool. Securing the CPU cores and IP (Intellectual Property) models is essential and the first stage to build the system level simulator. ARM usually provides their CPU models (ARM7, ARM9, ARM11 families) and standard system IPs (PrimeCells, such as PL300, PL340 and PL080) for the ARM RealView SOC designer, so it is relatively easy to get the ARM CPU and general PrimeCell IP models. For the SAVm IV simulator design, the major issue related to model securing is to get the StarCore DSP (SC2400) model and our proprietary data processing IPs working with the StarCore DSP for the multimedia processing.

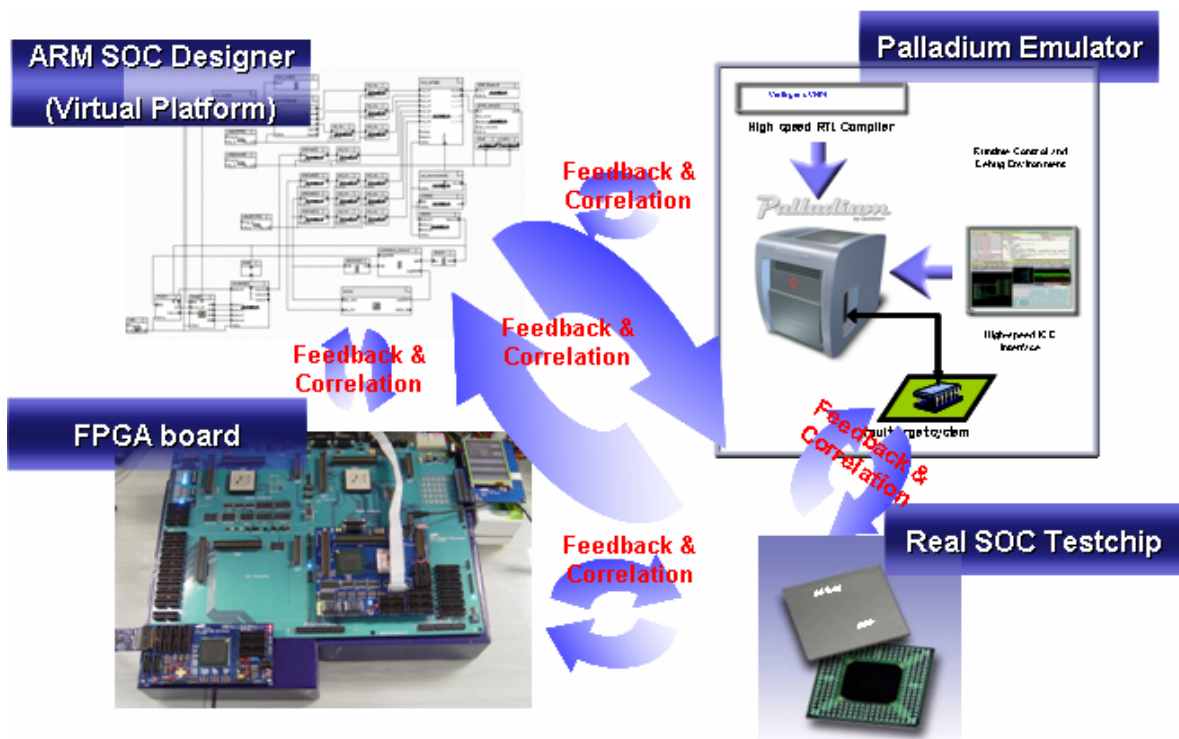


Figure 2. Various levels of architecture exploration environments

We also built a hardware emulation environment using the Cadence Palladium™ design verification system to evaluate the functional correctness and to verify the system performance [4]. The Cadence Palladium™ supports both simulation acceleration and in-circuit emulation, which speeds up for the verification about 100 to 10,000 times faster than RTL simulation. We developed an FPGA prototype board for the performance verification with a realistic image size and number of frames. The configuration of the FPGA prototype board is as follows. It has two daughter boards for each processor, ARM1136 and SC2400. It can run an MPEG-4/H.264 decoder software and display the output to LCD similar to the system level simulation environment. The board has an ARM11[5] board equipped with an ARM1136 CPU testchip and a StarCore[6] board having an SC2400 testchip and an 8M gate FPGA. The FPGA mother board has a CPU interface, DSP interface, two 8M gate FPGAs, SDRAM/DDR, memory port for OneNand memory, 2.2" LCD, SD, Sim Card, Keypad, Modem, etc. The FPGA contains the logics such as AHB/AXI (Advanced eXtensible Interface) buses, a static memory controller, a dynamic memory controller, an UART, an LCD controller, a post-processor, DMA Controllers, and a CommBox.

We also developed performance analysis unit (PAU) for the monitoring the performance characteristics including bus transactions [7]. This PAU can be used in FPGA boards and real chips. The PAU is designed for monitoring both the events related to Bus-Transaction (BTE) and the events related Bus-Contention (BCE). The BTE includes all events which are generated in a specific bus, such as transaction count, transaction data size, read/write latency distribution, and so on. On the other hand, the BCE contains events that are generated by the connection with several buses, such as contention between master or slave devices. The PAU is composed of two independent modules, Bus Monitor (BM) and Contention Monitor (CM). The BM measures BTE on a master bus, and the CM measures BCE on AXI interconnect. The CM is further divided into Master Contention Monitor (MCM) and Slave Contention Monitor (SCM) according to contention

devices. To increase the effectiveness of the PAU, we use the distributor for connecting buses with Bus (or Contention) Monitors flexibly. Figure 3 shows the overall structure of the PAU.

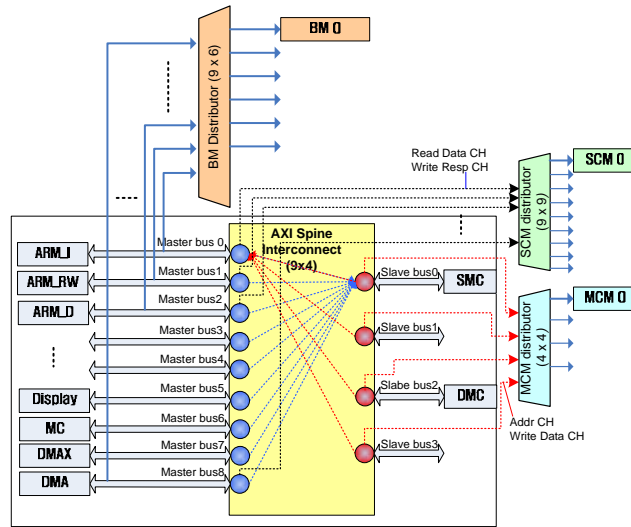


Figure.3 Overall structure of PAU for 9x4 AXI interconnect.

3 Considerations for Building and Utilizing These Environments

There are many factors to be considered to build and utilize these environments. Those factors include the simulation speed, accuracy, visibility and easiness of modification. Table 1 shows the comparisons of these factors for various architecture exploration environments we are building. The simulation speed based for the ARM SOC Designer, Palladium and FPGA board are the measured value based on our SAVm IV SOC implementation.

Table1 Comparisons of Various Architecture Exploration Environments

	ARM SOC	Palladium	FPGA	Real Chip
Simulation Speed (CPS)	10 – 30K	~130K	10 – 20M	100- 200M
Cycle Accuracy	High	Almost same	Very high	Same
Visibility/ Gathering	Very high	Low	Medium	Very Low
Easiness of Modification	Very easy	Difficult	Difficult	Almost impossible

Considering these characteristics of the architecture exploration environments, we use the ARM SOC designer to exploring various architectural alternatives based on its flexibility and visibility. After we refine the architectural alternatives with the system level simulator, extensive simulation using very huge input sets can be performed with the FPGA board and the Palladium hardware emulator to determine the optimal design parameters with huge amount of simulation for refined sets of architectural alternatives by the ARM SOC designer. Because the hardware emulator has almost same configuration and cycle accuracy with the actual design, it is very helpful for the final decision. The most important advantage of the FPGA board is very high speed of the execution. To improve the visibility and gather various statistics rather than a simple cycle counts, we developed and embedded the performance analysis unit (PAU) logic in the

FPGA. We also implemented this PAU logic into real SAVm IV SOC design to monitor the performance parameters within the real chip after we fabricate the chip. The PAU is expected to be helpful for the chip testing as well.

Another very important concern is the required efforts to build each environment. We can expect that the abstraction level of the model determines the efforts to build it as usual. That means the ARM SOC Designer which has highest abstraction level can be built with smallest efforts. It is only partially true in many cases in the actual SOC architecture design. This is because the transaction level models (TLM) for major IPs are usually not available in the early stage of the design. Contrary to the conventional expectation, RTL (Register transfer level) model for the major IPs are available before the high level system level simulation model (transaction level model). Many legacy IPs don't have transaction model yet because system level simulator are in adopting stage in these days. So, we should utilize the existing RTL assets in the FPGA board and Palladium hardware emulator with the PAU to explore the SOC architecture in the early stage in addition to the ARM SOC Designer. The VTOC [8] and Carbon [9] tools seem to be very promising because those help to secure the transaction model from the existing RTL to be used in the system level simulator like ARM SOC Designer.

The feedback and correlation are also very crucial for the future use of these environments. This is mainly because it is almost impossible to build very accurate system level simulator model due to the many limitation of it. So, some adjustments and guides to analysis of the simulation results generated from the system level simulator based on the correlation with the real chip and the system level simulator is very essential for the SOC architect to design derivative SOCs based on the SAVm IV architecture. We are performing these correlations with the system level simulator and other more low level environments, such as the Palladium and the FPGA prototype board.

[1] AMBA AXI Protocol Specification v1.0", *March*. 2004.

[2] ARM, SOC designer with MaxSim Technology, information available at www.arm.com/products/DevTools/MaxSim.html

[3] CoWare Inc, CoWare Platform Architect Overview, information available at <http://www.coware.com/products/platformarchitect.php>

[4] Cadence Design Systems, Inc., Palladium QEL Reference Manual, Product Version 1.1.1 QSR3, January 2004

[5] ARM, ARM1136JF-S and ARM1136J-Stm Technical Reference Manual, r0p2, ARM DDI 0211D, 2003. 8

[6] StarCore LLC, SC1000-Family Processor Core Reference Manual, June 2004

[7] Hyun-min Kyung, Gi-Ho Park, et al. "Performance monitor unit design for an AXI-based Multi-Core SOC Platform," To Appear in the proceedings of ACM Symposium on Applied Computing, Mar. 2007

[8] Tenison Design Automation, Tenison VTOC Datasheet, 2006

[9] Carbon Design Systems, Inc., Carbon with ARM information available at <http://carbondesigntools.com/corpsite/solutions/index.html>