Multi-Project System-on-Chip (MP-SoC): A Novel Test Vehicle for SoC Silicon Prototyping

Chun-Ming Huang[1], Kuen-Jong Lee[2], Chih-Chyau Yang[1], Wen-Hsiang Hu[1], Shi-Shen Wang[1], Jeng-Bin Chen[1], Chi-Shi Chen[1], Lan-Da Van[3], Chien-Ming Wu[1], Wei-Chang Tsai[1] and Jing-Yang Jou[1]

[1] National Chip Implementation Center (CIC), National Applied Research Laboratories, Taiwan
[2] Department of Electrical Engineering, National Cheng Kung University, Taiwan
[3] Department of Computer Science, National Chiao Tung University, Taiwan

ABSTRACT

In this paper, we propose a novel SoC design methodology referred to as Multi-Project System-on-a-Chip (MP-SoC), which can integrate multiple heterogeneous SoC design projects into a single chip such that the total silicon prototyping cost for these projects can be greatly reduced due to the sharing of a common SoC platform. The design flows for the system architecture, individual IP blocks, as well as the logic and physical implementations of MP-SoC are explored. The isolation mechanism to prevent interference among the IPs and the arbitration mechanism to grant the bus usage for master IPs are also presented. A test chip named MP-SoC-I that includes 8 SoC projects from 4 universities was selected as a demonstration example for verifying the MP-SoC design concept. This chip is designed and implemented in TSMC 0.13μm CMOS generic logic process technology, and the total silicon area for MP-SoC-I test chip is 4950μm×4938μm. Experimental results of MP-SoC-I test chip show that all projects are successfully implemented in the common platform and 82.91% silicon area is saved with this MP-SoC methodology as compared with the case where multiple SoC projects are fabricated individually.

I. INTRODUCTION

With the fast advance of IC fabrication and electronic design automation (EDA) technologies, the System-on-Chip (SoC) design concept has become more and more practical. An SoC can integrate a complex system into a single chip and achieve lower power, lower cost and higher speed than the traditional board level design. Among the existing SoC design methodologies, the platform-based methodology [1] is the most off-the-shelf one, where a platform is defined as an architectural framework consisting of a set of pre-qualified software and hardware IPs that are integrated into some specified on-chip connection architecture. Based on a well defined and verified SoC platform, even a small design team composed of graduate students can design and implement a complex SoC because the team members only need to focus on the creation of function-specific IP blocks and related embedded software.

Although the platform-based SoC design methodology is very helpful for academic SoC related research, due to the high fabrication cost, it is not easy to provide silicon prototyping opportunity for academic SoC design projects. The Multi-Project Chip (MPC) service [2]-[4] provided by many fabrication service institutions such as CIC, CMP, and IDEC, can effectively reduce the mask tooling and chip fabrication cost by merging multiple chip design projects into a single mask so as to share the high fabrication cost. However, the MPC concept is no help to reduce the high fabrication cost for the large silicon area demanded by each individual SoC.

Instead of verifying an SoC design project through silicon prototyping, virtual prototyping via hardware/software co-simulation environments such as Mentor Graphics Seamless CVE and CoWare ConvergenSC, or rapid prototyping via embedded processor-included field programmable devices such as Xilinx Virtex II, Altera Nios II, and Aptix System Explorer MP4CF, are two common solutions, but both of these two solutions cannot achieve the silicon-proof results of real chips.

In this paper, we propose a novel Multi-Project System-on-Chip (MP-SoC) design methodology that can integrate multiple SoC design projects that share a common SoC platform into a single chip. Since the common SoC platform (typically includes embedded processors, on-chip memories, on-chip connection architectures, peripheral devices and I/O pads) is shared by all SoC projects, only one common SoC platform has to be fabricated and thus the total fabrication cost can be dramatically reduced.

The rest of this paper is organized as follows. Section II expresses the MP-SoC vehicle design concept. In Section III, a complete MP-SoC design methodology is proposed. Then, a test chip called the MP-SoC-I chip is designed and implemented to demonstrate our MP-SoC design concept in Section IV, followed by a presentation of the chip results and our test plan in Section V. Finally, we conclude this work in Section VI.

II. MP-SoC VEHICLE DESIGN CONCEPT

The key idea of the MP-SoC methodology is to reduce the total chip area by sharing the common resources such as CPU/DSP processor, on-chip bus, embedded memory, peripheral devices and I/O pads among the projects to be integrated into a chip. The saving of the chip area can be formulated as follows.

Assume that there exist N SoC projects. The total area for independently implementing the N SoC projects and by using our MP-SoC methodology, denoted as A_{total} and A_{MP-SoC}, can be formulated as Eq. (1) and Eq. (2), respectively.

$$A_{total} = N \times A_{shared} + \sum_{i=1}^{N} A_{IP,i} + \sum_{i=1}^{N} A_{overhead1,i} \qquad (1)$$

$$A_{MP-SoC} = A_{shared} + \sum_{i=1}^{N} A_{IP,i} + \sum_{i=1}^{N} A_{overhead2,i} \qquad (2)$$

where A_{shared}, $A_{IP,i}$, $A_{overhead1,i}$, and $A_{overhead2,i}$ indicate the common component area that can be shared by all SoC projects, the dedicated IP's area for the *i*-th SoC project, the overhead area due to the integration of dedicated IPs of the *i*-th project to a dedicated SoC, and the overhead area due to the integration of dedicated IPs for the *i*-th project to the MP-SoC chip, respectively.

Applying this vehicle mechanism, the area cost saving can be advantaged from the reduction of *N*-1 A_{shared}. Here, we define a cost evaluation index R_{saving} for the proposed MP-SoC vehicle as follows:

$$R_{saving} = \frac{A_{total} - A_{MP-SoC}}{A_{total}}$$

$$= \frac{(N-1) \times A_{shared} - \sum_{i=1}^{N} (A_{overhead2,i} - A_{overhead1,i})}{A_{total}} \qquad (3)$$

Clearly the larger value in Eq. (3) is, the more fabrication cost saving can be anticipated.

III. MP-SoC DESIGN METHODOLOGY

In this section, we derive the most complete design flows that are users friendly for quick hardware and software development and verification without sacrificing development time and resulting performance.

Overall MP-SoC design flow: The platform based design methodology is adopted for developing this MP-SoC chip. However, many problems and challenges exist if the current IC/SoC design flow is directly applied. In order to save design time, we have developed a set of new design flows for MP-SoC. The developed design flows include a system architecture design flow, an IP block design flow, a logic implementation design flow, and a physical implementation design flow. These design flows greatly simplify the design and verification of the MP-SoC system while seamlessly integrating industry-standard EDA tool environments.

Fig. 1 shows the high-level view of the MP-SoC system design flow. At the beginning, the test environment planning and architecture of the system are specified. It is in this stage that the test planning is decided, the basic hardware components are identified, and the component interfaces, including data and control signals, are fixed. Next, we create an implementation platform and the whole chip verification environment. The system and the IP specifications are obtained as well. The system/IP specifications and implementation platform are then used for IP development. In the IP block design stage, we ask the universities joining the MP-SoC project to follow our proposed IP block design guideline and verify their IPs using our proposed verification environment. After all the IPs are designed and well verified in our verification environment, we

gather all the IPs into our MP-SoC system and perform logic implementation according to the system specification. The physical implementation is finally completed and the resulting circuit is taped out to the TSMC foundry.
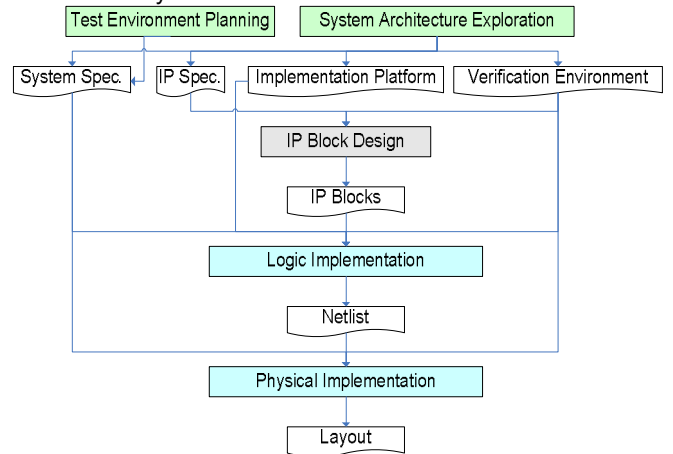

Fig. 1. Overall MP-SoC design flow

System Architecture Design Flow: To offer the verifying system-level performance and features, we create a system architecture design flow to further improve verification time benefits for the following IP designs. First, we investigate the specifications of all IPs to determine the IP requirements such as the memory space for slave IPs, internal memory size and the number of external pins. Secondly, the system memory map for all AHB slave IPs and the arbitration mechanism for all AHB master IPs are decided according to the system performance and design complexity. Finally, the MP-SoC platform, which consists of an implementation platform and a verification environment, is created. In the meantime, the system/IP specifications such as chip/IP IO pins and constraints are developed and used in the following IP, logic and physical implementation flows.

IP Block Design Flow: To integrate the IPs into our platform easily and smoothly, we create an IP block design flow. This design flow is used for universities who attend this MP-SoC project. Students start with their own IP design according to their functional specifications. To integrate the IP to MP-SoC implementation platform, an AHB wrapper should be realized based on our MP-SoC platform constraints. Each IP design is then delivered to the logic design flow.

Logic and Physical Implementation Design Flows: The goal of these two implementation flows is to ensure that this complex SoC system can be successfully realized. After all IPs' RTL designs from universities are gathered, the whole chip RTL simulation is performed to ensure the correct functions. The whole chip design is synthesized according to the whole chip constraints, and then static timing analysis and gate-level simulation are performed. The pre-layout gate-level netlist is then delivered to physical implementation design flow. The layout is produced by a P&R tool. The RC extraction, static timing analysis, and whole chip post gate-level simulation are performed, with DRC and LVS also done
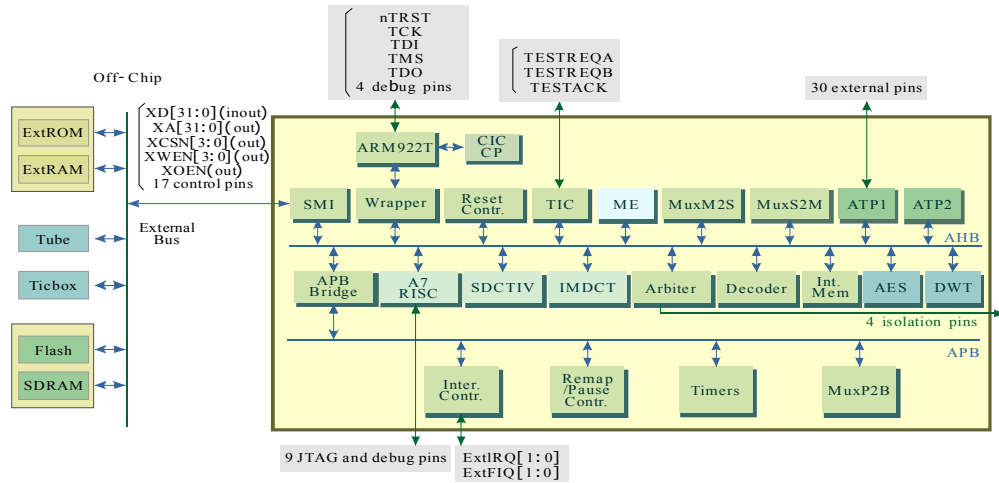
Fig. 2. MP-SoC-I block diagram

to ensure the layout correctness. Finally, the GDSII is taped out to the TSMC foundry.

In summary, the design flows described in this section not only provide an environment for IP creation and integration but also seamlessly leverage the various state-of-the-art EDA tools to accomplish the necessary design tasks.

IV. DESIGN and IMPLEMENTATION of MP-SoC-I TEST CHIP

Based on the design flows created in Section III, designers can easily sketch their design concepts, use EDA tools and the design flows, and tape out chips to accomplish an MP-SoC system design. To further demonstrate the efficiency of our MP-SoC design concept and methodology, we realize a test chip, referred to as *MP-SoC-I*. Fig. 2 illustrates the block diagram of this chip. It is constructed with the ARM AMBA bus architecture [5] in which it contains an ARM high performance bus (AHB) for high performance devices and an ARM peripheral bus (APB) for low cost peripheral devices. The main components of the AHB bus include an ARM922T CPU core, some internal memory, and a TIC module, while the main components of the APB bus are a timer, an interrupt controller, and a remap/pause controller. The communication protocols between the MP-SoC chip and the off-chip devices are the external memory interface, debug interface, interrupt and some control signals. In addition, there are two kinds of off-chip memory systems provided in our MP-SoC system. One consists of flash and SDRAM memory, and the other consists of ROM and SRAM memory. Users can select the suitable memory interface for their SoC projects.

There are 4 universities attending this MP-SoC-I project. The IPs from these universities include an AES engine for communication systems, a DWT engine for image compression, a RISC processor (A7 RISC) for system control, a SDCTIV and a IMDCT engine for MP3 applications, two advanced test platforms (ATPs) for SoC testing, and a motion estimation (ME) engine for video compression. This MP-SoC-I platform is thus capable of handling various applications, such as communication, image and video/audio systems.

A. Isolation and Arbitration Mechanism

Since there are many kinds of IPs designed by different universities to be integrated into the MP-SoC-I chip, appropriate isolation and arbitration mechanisms are needed for all devices so that the no interference among IPs will occur and performance of any single IP will not be degraded

Since the master IPs can issue the read or write requests to slave IPs, the interference from master IPs' inappropriate request should be taken into consideration. In MP-SoC-I, there are 6 masters on AHB bus and the priority lists from high to low are TIC, pause controller, ATP1, ATP2, A922T and A7 RISC. In order to avoid the bus interference from the dedicated master IPs of each project and the ARM922T master IP, we present an isolation mechanism as described in Fig. 3. It is made up of four sets of 2-to-1 multiplexers, which are used to decide whether the master IPs are enabled by using 4 external isolation pins ATP1_EN, ATP2_EN, A922T_EN, and A7_EN.

As for the arbitration mechanism, in general four basic types of arbitrator algorithms can be used: Fixed, Round Robin [6], Lottery, and TDM [7]. Although the Round Robin, Lottery, and TDM algorithms can provide a better performance, we still adopted the Fixed architecture in MP-SoC-I due to its simplicity and better success opportunity for this first MP-SoC design.
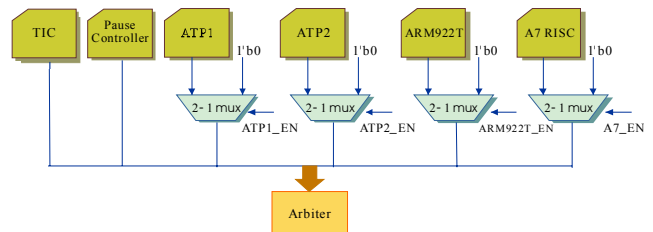

Fig. 3. MP-SoC-I isolation mechanism

B. The Implementation Platform and Verification Environment

For the MP-SoC-I chip, we create an implementation platform and a verification environment to accelerate the design, implementation, verification, and integration processes. In the implementation platform, we reserved the empty blocks for universities IPs' integration, each design team needs only to connect their IPs to the reserved empty blocks, and follow the system map planning to complete their application program. This implies that this is a plug and play platform. This platform in conjunction with the developed flows results in a very time-efficient design and implementation environment for individual IPs to be put in the MP-SoC chip.

The MP-SoC-I verification environment is developed to fulfill the need for heterogeneous applications to be verified quickly. The application programs written in C/C++ and assembly code are first respectively compiled by a C compiler and an assembler to obtain the object codes. A linker is then used to link all the object codes to produce an executable file. A format conversion utility is adopted to convert the binary executables to an appropriate memory format, such as flash or ROM memory. All the software including instructions and data are initially stored in flash or ROM memory. After the system reset signal is activated, the MP-SoC-I chip starts to work. Users can observe the debug display messages via the Tube display module and memory interface.

V. CHIP RESULTS and TEST PLAN

The MP-SoC-I chip was fabricated in TSMC 0.13μm 1P8M logic process. Fig. 4 shows the MP-SoC-I chip photo and Table 1 summaries the features and measured characteristics of MP-SoC-I. The resulting core size is about 3700×3700 μm^2 and the overall chip size including I/O pads is 4950×4938 μm^2. The total number of I/O pads is 256, which consists of 104 power pads and 152 signal pads. From the layout in Fig. 4, it can be seen that the ARM922T CPU occupies a large fraction of the total chip area. Based on Eq. (3), we find that 82.91% silicon area is saved when adopting the MP-SoC concept compared with the case that multiple SoC projects are fabricated individually.

This MP-SoC-I chip is tested via Agilent 93000 ATE. When the chip is measured on ATE, the input patterns are those captured in advance during the Verilog simulation. The chip output results captured by the ATE are then compared with the expected patterns from the Verilog simulation. The functional testing with low-frequency (100 MHz) was done in the worst case condition. The other way to measure the MP-SoC-I chip is through the system development board which is under development.

Table 1 MP-SoC-I chip characteristics

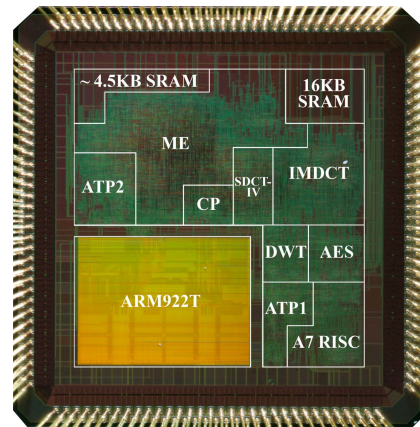| Critical Delay Time	10 ns
Number of I/O Pads	256
Core Area	3700μm x 3700μm
Chip Area	4950μm x 4938μm
Process Technology	TSMC 0.13μm CMOS

Fig. 4. MP-SoC-I chip photo

VI. CONCLUSIONS

Low cost silicon prototyping techniques are very helpful for academic SoC design projects. In this paper, a novel design concept, Multiple-Project SoC, is proposed. It can integrate multiple SoC projects into a single chip by sharing a common platform to reduce cost. To integrate the multiple SoC projects into our common platform easily, design flows for the system architecture, individual IP blocks, and logic/physical implementation are provided. A test chip named MP-SoC-I which includes 8 SoC projects from 4 universities was selected as a demonstration example for verifying the MP-SoC design concept. This 4950μm×4938μm MP-SoC-I test chip is implemented in 0.13μm TSMC CMOS technology. Measured results reveal that our MP-SoC-I test chip can save 82.91% silicon area as compared to the individually fabricated chips. Based on the experimental results, we conclude that this MP-SoC design concept is very helpful for academic SoC related research since it greatly enhances the silicon prototyping opportunity for academic SoC design projects.

REFERENCES

[1] H. Chang et al., "*Surviving the SoC Revolution: A Guide to Platform-based Designs,*" Kluwer Academic, Norwell, Mass., 1999

[2] J. S. Hwang, "Multi-Project Chip Service for University and Industry in Taiwan," in *Proc. 1997 IEEE Asia and South Pacific Design Automation Conference*, 1997, pp. 359-363

[3] B. Courtois, "MPC Services Available Worldwide," in *Proc. 1994 IEEE Asia-Pacific Conference on Circuits and Systems*, 1994, pp. 266-275

[4] C. M. Kyung et al., "Multi-Project Chip Activities in Korea-IDEC Perspective," in *Proc. 1997 IEEE Asia and South Pacific Design Automation Conference*, 1997, pp. 353-357

[5] ARM Ltd., "AMBA Specification Revision 2.0," May 1999

[6] E.S. Shin, V.J. Mooney III and G.F. Riley, "Round-robin Arbiter Design and Generation," in *Proc. 2002 International Symposium on System Synthesis* 2002, pp.243-248

[7] K. Lahiri, A. Raghunathan, G. Lakshminarayana, "LOTTERYBUS: A New High-Performance Communication Architecture for System-on-Chip Designs," in *Proc. 2001 IEEE/ACM Design Automation Conference*, 2001, pp.15-20