

Multi-Core Design Automation Challenges

John A Darringer

IBM T. J. Watson Research Center
P. O. Box 218
Yorktown Heights, NY 10598
1-914-945-2742
jad@us.ibm.com

ABSTRACT

The trend to multi-core chip designs presents new challenges for design automation, while the increased reuse of components may offer solutions. This paper describes some of the key challenges with attention paid to three enablers: a physical architecture to streamline chip integration, the linking of early analysis tools around shared data and an updated verification approach for multi-core designs.

Categories and Subject Descriptors

B.2.2 Performance Analysis and Design Aids, C.4 Performance of Systems, J.6 Computer-Aided Engineering.

General Terms

Design, performance, verification.

Keywords

Multi-core systems, performance analysis, power estimation, floorplanning, thermal analysis, verification, design automation.

1. THE MULTI-CORE TREND

IBM's introduction of the first multi-core processor chip, Power4 in 2001 [1], was an important step forward. Multiprocessing was commonplace, but by placing 2 processors with a shared cache on one chip, the designers were able to achieve much greater communication bandwidth and resulting performance. IBM next introduced the Power5 chip in early 2004 [2], the first multi-core processor with simultaneous multithreading. Enabling four active threads on one chip greatly increased resource utilization along with performance.

Today, nearly all processor chips use multiple cores in an attempt to deliver more system performance within their power constrained environment. Operating systems have been able to adapt to the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

DAC 2007, June 4–8, 2007, San Diego, California, USA.

Copyright 2007 ACM 978-1-59593-627-1/07/0006 ...\$5.00.

multi-core design so far, but some important applications are not easily modified to exploit multiple cores for increased performance. As the number of cores and threads per chip increase, it will be the ability to restructure legacy software and create new parallel applications that will be needed to maintain the historic growth in system performance with multi-core systems.

2. DESIGNS FOR EMERGING MARKETS

A related trend is the appearance of multi-core designs with heterogeneous cores that can provide striking performance for particular applications. One example is the Cell chip that was developed for the video game market [3]. The combination of the Power PC control processor and 8 specialized engines sharing a high-bandwidth interconnect has demonstrated significant advantages over more traditional designs. It is certain that system designers will find similar ways to mix conventional processors with diverse accelerators aimed at specific tasks, and feed them with a communication network and memory hierarchy to provide equally remarkable results in many important areas, such as data base transaction processing, search, medical, security, and media processing. This is an exciting and challenging period for system architects.

3. BEYOND ASIC SoC

In one sense, multi-core chips are not that new. ASIC providers such as NEC, IBM, ST, and TI have been providing multi-core SoCs for years. In fact, IBM and Cisco teamed to build a 172-core 40-terabit router chip [4]. The combination of a rich library of components designed for reuse, a scalable interconnect fabric and a highly automated implementation methodology makes it possible for designers to rapidly take advantage of silicon's integration ability and deliver extremely competitive products to market in a short amount of time.

However, when the ASIC approach is contrasted with that used in processor design, it is apparent that even further performance and power efficiency can be gained through increased attention to chip organization and interconnects at all levels combined with custom circuit design.

If a new multi-core design methodology could be developed that delivered processor-like power performance with ASIC-like efficiency, it would enable system designers to go after the many new and emerging markets with innovative multi-core designs. What are the key challenges in creating this new modular multi-core methodology?

4. THREE ENABLERS

One critical difference between the processor and ASIC approaches is the extreme attention paid to interconnect early and throughout processor design. For ASIC design, IP blocks have a well defined interface that allows them to easily connect to a bus or other communication network, but typically they do not have a fixed layout. Instead these “soft” blocks go through synthesis, placement and wiring each time they are used. This provides valuable flexibility in their use, but it also leads to some uncertainty and variability in the final performance. ASIC circuits are designed to ensure correct behavior in a wide range of cases and as a result, their performance and power efficiency may not always be optimal. To achieve better power efficiency the circuit and interconnect layout needs to be considered in advance and designed in a modular parameterized fashion that can achieve top performance in all anticipated instantiations. This means that in addition to a logical architecture that allows IP blocks to be interconnected in a nearly arbitrary fashion, there needs to be a *Physical Architecture* that ensures that no matter how blocks are configured, the resulting layout will achieve predictable and processor-like results.

A second major difference in design approaches is that processor teams spend much more effort in the early stages of conceptual system design. Certainly, ASIC teams, in general, do develop functional models to confirm expected behavior and to enable early software development. They also employ cycle-accurate models to predict hardware performance and floorplans to estimate critical delays. But, processor teams typically invest even more effort in this process. They have a team of experienced experts that work closely to consider system tradeoffs in the early “concept” phase of design. They use multiple complex performance models, comprehensive floorplans and spreadsheets together with detailed implementations of selected sections, all to predict attainment of critical design criteria. While today’s teams do an excellent job, this is a process that is very dependent on seasoned experts and does not scale well to address multiple new markets and the dramatically growing list of choices facing system designers: Which cores and accelerators to use? What interconnect and memory hierarchy will yield the best result? What I/O bandwidth is needed and can be achieved? How should the many technology options be exploited, such as voltage islands and variable frequencies to maximize power performance? There is an opportunity to bring automation to this early stage of design and provide a needed advance in productivity through *Integrated Early Analysis*.

Finally, a third enabler to multi-core design is an updated *Multi-Core Verification Process*. The verification of large multiprocessor systems has always been one of the most demanding challenges in EDA. Now those complex systems are moving to single chips with reduced observability. As technology options are made available, such as power gating, variable frequencies and asynchronous communication the verification task is compounded. The key is to take advantage of the reuse that is inherent in multi-core design.

5. PHYSICAL ARCHITECTURE

The central idea is to establish physical modularity, as well as logical modularity, through a library of reusable blocks that have both a logical and physical specification. As these blocks are composed to provide a needed function, each block has associated

with it a parameterized layout or layout generator that produces a highly efficient circuit and interconnect layout. Accompanying the library is an overall chip physical architecture that anticipates the physical integration of these blocks and streamlines the complex task of final chip integration. This vision carries certain requirements:

1) The blocks need to be implemented in the same specific technology. Technology migration aides can help with shifts to new technologies.

2) The blocks need to access power in a consistent manner. This is complicated by the growing use of voltage islands for power management and the importance of minimizing voltage fluctuations. A scheme is needed that provides the correct balance of flexibility and predictability without significant human effort for each new design.

3) There needs to be a standard method for distributing and controlling clocks to the blocks. This is especially difficult when performance and power efficiency are key design criteria. One approach is for each block to ensure that delay and skew requirements are met within the block. This is established as part of its physical layout. Still there needs to be an automated or semi-automated process for distributing and controlling clocks to the blocks for clock gating and local frequency control. During chip integration buffer insertion and clock routing is performed to meet the specified requirements at each block.

4) There needs to be a consistent way to access chip I/O drivers, which has to be planned for in the physical architecture to avoid disrupting the other competitors for wiring capacity.

5) There needs to be a uniform method for adding test facilities and bringing up the chip after fabrication.

6) Finally, and probably most importantly, each block must have a standard method for communication. Whether through a bus, network or point-to-point links, the architecture needs to plan for all communication in all configurations and in a way that can meet the high bandwidth and low latency requirements of a multi-core design.

The goal of the physical architecture is to greatly simplify chip integration, while delivering competitive performance and power efficiency. A vital companion to this architecture is a tool to carry out the chip integration. The more the tool can recognize suboptimal layouts and avoid or correct them with or without designer guidance, the more robust the overall approach will be. The requirements above are daunting, but such an approach to physical design is critical for the benefits of multi-core design to reach the many opportunities available.

6. INTEGRATED EARLY ANALYSIS

Early in the design of a processor, a small team of experts in performance analysis, power estimation, chip layout, packaging, bus architectures and memory hierarchies are brought together to consider as many alternatives as possible to meet a target set of product requirements. They typically use spreadsheets and personal programs to form their estimates. Assumptions, constraints and results are shared, but informally. Within each of these disciplines some improvement is needed and can be made to meet the increased demand of multi-core designs. But an even greater impact can be made to the overall productivity of early

analysis, through the tighter integration of the tools used via a data base to share assumptions and results.

6.1 Performance Modeling

Early analysis begins with a performance model – a model that embodies a great many assumptions about the design architecture, the technology, the packaging, and market requirements. Performance models can be analytic, trace-driven or execution-based. All have served processor teams well. But as designs with a growing number of diverse cores are considered for new markets, each of these methods is stressed. Not only must the core be modeled with sufficient accuracy to evaluate subtle micro-architectural ideas, they must be fast enough to consider large numbers of cores operating on a shared network. A team at Delft is exploring new analytic models for memory hierarchies for multi-core designs [5]. The Mambo group in IBM is using a combination of fast high-level modeling with more precise models to handle large numbers of cores [6]. Also many groups are looking at SystemC transaction level modeling to attack this problem [7].

6.2 Power Estimation

Power has become the dominate design criteria and its accurate prediction is especially important and difficult at the early stages of design. Even when a design has an RTL specification or a lower level implementation; it is still difficult to accurately estimate power dissipation, because of the sensitivity to variation in workloads, as well as the degree of clock and power gating. In the early stages, a design may consist only of an interconnected set of components that do not have an RTL embodiment. In this case, it is common to form estimates by applying scaling to a detailed analysis of similar existing designs. The method works well for incremental advances to existing chips, but loses accuracy when new hardware components are used with new technologies.

With each core running a different stream of instructions there can be a wide variation of on chip power distribution leading to voltage droop and local dynamic hot spots. To enable designers to access exposures and evaluate alternative power management schemes, power estimation methods need to deliver results dynamically on a cycle-by-cycle basis. This requirement is very important and different from predicting average power dissipation.

6.3 Floorplanning

The goal of a Physical Architecture is to greatly simplify the task of integrating pre-designed or generated components into a multi-core chip. It will provide a “master plan” that anticipates configurations and limit options. Still there is a need to analyze any specific layout to confirm that all assumptions about power, clock and I/O routing are met and that all critical interconnects meet required latencies. Different configurations can effect congestion and the potential for coupling noise. Arrays are a major portion of today’s chips and often require special treatment for power or I/O access that can complicate floorplanning. The physical architecture needs to plan for on-chip memory and arrays, as well as their access to any shared communication network. It is important for the early floorplan to be comprehensive. Area, pins and other resources must be

anticipated for all on-chip functions, even though some may be far from complete implementation in the conceptual stages of design.

6.4 Thermal Analysis

In a multi-core design each core can be running a different mix of instructions leading to a wide variation in dynamic power dissipation and in on-chip local temperatures. To be able to predict this thermal behavior requires an accurate analysis of the dynamic power dissipation of all on-chip components, an accurate floorplan and package model plus a dynamic thermal modeling tool that can incorporate what is known in the early stages about chip and package thermal properties, while being fast enough to provide prompt feedback in early tradeoff studies. Often thermal issues are identified late in a chip’s design when correction is very expensive.

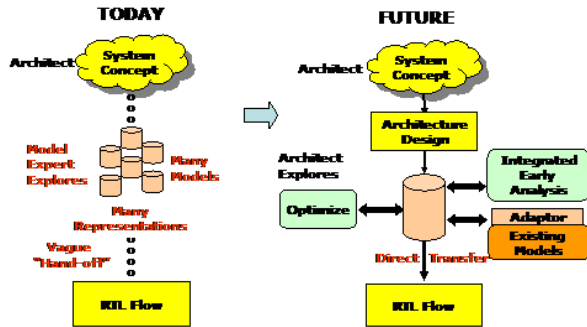
6.5 Integration

As discussed above, the trend to multi-core designs is stressing the individual analysis methods used in the early stages of design and they each can be improved. But there is also a significant productivity gain that can be realized by more closely integrating the separate tools around a shared data base. The growing importance of power estimation is forcing a closer coupling of power estimation models with performance models. The simple approach of exchanging files limits the ability to analyze dynamic power. The linkage between performance models and a chip’s floorplan is typically quite loose. Of course, performance models can and do incorporate latency information about memory accesses, network communication and global interconnects. But it is typically a manual process to extract these numbers from a floorplan and enter them in a performance model. Automating this step could enable more productive tradeoff analysis and enable the search of a much larger design space. Similarly, thermal analysis is often done with selected data from power models, floorplans, technology and packaging – data all gathered manually, creating many opportunities for transcription errors or misunderstanding.

6.6 Direct Hand-off

The models used for early analysis contain much valuable information that is often communicated in a written document or verbally. Transferring these decisions, assumptions, constraints and predicted results directly to the subsequent tool flow can greatly speed implementation. Moreover, with the modular approach of multi-core design, the proven implementations of existing components can be assembled to provide a significant portion of the chip implementation. The physical architecture provides rules for guiding the assembly and for integrating the remaining parts of the chip. New components still have to be implemented, but they have a more complete specification and a detailed context to simplify this task.

Figure 1 contrasts the informal manual process often used today for early analysis with the vision of Integrated Early Analysis. Instead of many separate models each with their own input formats and files, all assumptions, constraints, technology parameters and analysis results would be stored in a shared repository. Analysis tools can operate concurrently and share data in memory to avoid inefficient file I/O. Having centralized data and a common method



for invoking analysis tools sets the stage for exploring automated optimization. Adaptors can accommodate tools and models that may be more difficult to integrate. Finally, design data and constraints can be passed directly to downstream tools to guide the RTL implementation. Tightly integrated analysis has been proven to be essential for RTL design. Similarly, coupling the tools of early analysis will be vital for architects to find the best multi-core solutions in the future.

7. MULTI-CORE VERIFICATION

Verification has always been the dominate challenge for large systems; and while multi-core chips present many new demands, they also offer some potential solutions. The key is reuse. The challenge remains of verifying that a block performs according to a specification, but the task of verifying the correct behavior for the multi-core chip is becoming a problem of verifying that the communication network operates correctly and that the blocks connect to it as required.

For ASIC SoC designs most IP blocks are well tested and debugged through multiple uses. In addition, there are strict rules for connecting the blocks and well defined communication networks that limit the complexity of assembled systems. As a result, for the configurations within a planned range, system verification is often reduced to verifying that the blocks are correctly connected to the communication network. Formal methods have been applied to this problem to prove safety properties for an ASIC SoC [8]. However, with multi-core processor designs, the communication networks are typically more complex, for example, offering facilities to preserve cache coherency or allowing special bypass options for performance reasons. It is here that there is an opportunity to extend verification methods.

High-level modeling is a clear direction and has been applied in the verification of memory protocols. In one example [9], an abstract specification was developed as part on the design for a complex cache coherency protocol and used with an explicit model checker to expose problems and establish safety properties for an IBM product.

Still today, extremely complex, hierarchical protocols are beyond the reach of model checking, even using abstract models and new approaches are being studied to partition unmanageably large model checking problems into a set of smaller, solvable problems [10]. Assume-guarantee methods show promising results when applied to these complex multi-core protocols [11]. The combination of higher-level specification methods and improved proof methods are the key to simplifying the verification of multi-core design built with pre-verified blocks.

8. SUMMARY

A long history of innovation in design automation has enabled designers to take advantage of silicon advances and deliver a remarkable growth in system performance. As designers move to exploit multi-core designs for more power efficiency, new DA innovations will be required again. This paper focused on three key enablers:

1) A physical architecture is needed to parallel the logical architecture that is the basis of today's ASIC SoC methodology. This physical architecture must anticipate possible configurations and carefully plan for all levels of interconnection for communication, clocks, power, test, and I/O; in order to retain the efficiency of processor design, while gaining productivity from reuse.

2) Integrated early analysis can improve the productivity of design exploration that is critical to finding the best multi-core solutions. Linking the traditionally separate tools used in the architectural tradeoff analysis phase of design and sharing data in a common data base can avoid conflicting assumptions, streamline multi-objective analysis and yield a more efficient implementation process for multi-core designs.

3) Verification remains the largest effort in design and multi-core designs are stressing today's verification techniques. But the prospects of increased reuse can help, specifically with the verification of the final assembled chip. A multi-core verification process that operates at a high-level of abstraction and takes advantage of the module structure can greatly simplify the verification of the assembled system of pre-verified modules.

9. ACKNOWLEDGMENTS

A special thanks to Pradip Bose, Steven German, Ruchir Puri, Reinaldo Bergamaschi, Indira Nair, Nagu Dhanwada and Emrah Acar for their contributions to the ideas described here.

10. REFERENCES

- [1] J. M. Tendler, J. S. Dodson, J. S. Fields, Jr., H. Le, and B. Sinharoy, "POWER4 system microarchitecture", IBM Journal of R&D, Vol. 46, No. 1, 2002, pp. 5-26.
- [2] B. Sinharoy, R. N. Kalla, J. M. Tendler, R. J. Eickemeyer, and J. B. Joyner, "POWER5 system microarchitecture", IBM Journal of R&D, Vol. 49, No. 4/5, 2005, pp. 505-522.
- [3] D. Pham, Asano, S. Bolliger, M. Day, M.N. Hofstee, H.P. Johns, C. Kahle, J. Kameyama, A. Keaty, J. Masubuchi, Y. Riley, M. Shippy, D. Stasiak, D. Suzuoki, M. Wang, M. Warnock, J. Weitzel, S. Wendel, D. Yamazaki, T. Yazawa, K. "The Design and Implementation of a First-Generation CELL Processor". In ISSCC Digest of Technical Papers. San Francisco, CA, February 2005 pp. 184-5
- [4] J. Koehl, D. Lackey, G. Doerre, "IBM's 50M Gate ASICs", Proceedings of the ASP-DAC 2003. Japan, January 2003 pp. 628 – 634
- [5] A.L. Varbanescu, H. Sips and A. van Gemund, "Semi-Static Performance Prediction for MPSoC Platforms", In Proc. of the 12th International Workshop on Compilers for Parallel Computers (CPC 2006), A Coruna, Spain, January 2006.

- [6] P. Bohrer, J. Peterson, M. Elnozahy, R. Rjamony, A. Gheith, R. Rockhold, C. Lefurgy, H. Shafi, T. Nakra, R. Simpson, E. Speight, K. Sudeep, E. Van Hensbergen, and L. Zhang, [“Mambo: A full system simulator for the PowerPC architecture”](#), SIGMETRICS Performance Evaluation Review, March 2004, pp. 8-12.
- [7] Nagu Dhanwada, Reinaldo A. Bergamaschi, William W. Dungan, Indira Nair, Paul Gramann, [“Transaction-level modeling for architectural and power analysis of PowerPC and CoreConnect-based systems”](#), Design Automation for Embedded Systems, Vol. 10, No. 2-3, September 2005, pp. 105-125.
- [8] A. Goel, W. R. Lee, [“Formal verification of an IBM CoreConnect processor local bus arbiter core.”](#) In Proceedings of the 37th Conference on Design Automation, June, 2000.
- [9] S.M. German, [“Formal Design of Cache Memory Protocols in IBM”](#), Formal Methods in System Design, 2003, pp. 133-141.
- [10] S. Park and D.L. Dill, [“Verification of Cache Coherence Protocols by Aggregation of Distributed Transactions”](#), Theory of Computing Systems, 1998, pp. 355-376.
- [11] Xiaofang Chen, Yu Yang, Ganesh Gopalakrishnan, and Ching-Tsun Chou, [“Reducing Verification Complexity of a Multicore Coherence Protocol Using Assume/Guarantee.”](#) Formal Methods in Computer Aided Design (FMCAD), IEEE, San Jose, November 2006, pp. 81-88.