PANEL: SLIP - 10 Years Ago and 10 Years From Now

Patrick Groeneveld
Magma Design Automation, San Jose, CA, USA

Lou Scheffer
Howard Hughes Medical Institute, Ashburn, VA, USA

Dirk Stroobandt
ELIS Department, Ghent University, Gent, Belgium

Categories and Subject Descriptors

J.6 [**Computer-Aided Engineering**]: Computer-aided design (CAD)

General Terms

Design, Performance

Keywords

SLIP, Interconnect prediction, System-level design

Panel Summary

Founded in 1999, the ACM SLIP Workshop is now in its 12th year. The 2010 SLIP Panel session will highlight perspectives from three individuals who have had great influence on the course of SLIP, and provide an opportunity for lively discussion by workshop attendees of prospects for the next 10 years of SLIP. This panel summary records preliminary thoughts of the panelists on two starting questions.

Q1: How do you see the scope, role, etc. of SLIP having changed since 1999?

Patrick Groeneveld.

My job at Magma forces a low-level perspective on interconnect planning in an ASIC design flow. The chip needs to work, and timing and DRC must be closed by any means. Therefore, my interest in system-level interconnect prediction is primarily driven by the correlation accuracy as compared to the actual physical routing wires. Miscorrelation leads to routing and timing closure issues in the back-end. In fact, there is no meaningful application of system-level interconnect prediction without this correlation.

Since the first SLIP in 1999, the core problem statement has not changed much. We must admit that SLIP tool applicability has been rather marginal. SLIP is the project of a small community, and the practical use is constrained to several basic technology-level interconnect planning issues (e.g., how many metal layers and what should be their pitch?). Since 3-D chips do allow the user to control the structure of the interconnect layers, I would expect more interest in the coming years.

Lou Scheffer.

SLIP'10, June 13, 2010, Anaheim, California, USA.
ACM 978-1-4503-0037-7/10/06.

10 years ago, we were mainly concerned with wire length and statistical properties of a graph theoretical interconect model combined with a simple model of the physical characteristics of interconnect. The main correlation issue was between the detailed router and the interconnect estimates.

Now performance and power are critical, in addition to wireability. This makes SLIP much harder, since high performance interconnect takes more area and more power. A design, at low performance, might be wired with all minimum size wires. For the exact same design, with an unchanged interconnect graph and Rent descriptors, as the performance target is raised, more and more wires will need to be promoted to higher performance layers or technologies, and the wiring space will grow. Also, depending on the design, anywhere from just a few wires, to all of them, will need to be promoted. This shows interconnect prediction must take performance, function, and the timing details of the design into account.

Also, 10 years ago, there was very little interaction between coding of data and the design of interconnect, at least for the short range connections found on chips. Now this is increasingly important, particularly since crosstalk can account for a large percentage of delay.

Dirk Stroobandt.

My position statement: "Forget about individualism, it's all about socialism".

In a decade of SLIP research there has been a strong dichotomy between the fundamental statistical nature of Rent's rule based estimates and the quest for individual wire performance measures.

On the one hand, the very nature of Rent's rule (as a scaling argument) naturally leads to a global model of interconnection lengths. All wires are "connected" (pun intended), so that classifying one wire automatically restricts the classification of the other wires. The force-directed placement model illustrates this very well, as all gates are modeled as being connected by springs where you would have connections. Pushing two gates together releases the force in one spring, but increases it in other springs connected to that particular gate. This global view on the interconnect problem is also very well reflected by the Donath-based interconnection length prediction models and (especially) their extensions of the mid-1990's.

The global interconnection length model was very well suited to the CAD problems of that time, where placement was based on a global estimate (and optimization) of total wire length, and routing was mainly based on congestion maps which are already a bit more localized but still global

in the sense I discussed before. This has then resulted in a great deal of work in the SLIP community on placement optimization, routing and routability improvement, floorplanning, manufacturability and yield, technology extrapolations (especially for the ITRS technology roadmap), and even on-chip power distribution. Many of these models have also been extended to three-dimensional and (partly) optical systems.

On the other hand, however, during the 10 years of SLIP, it became clear that many problems are in fact related to individual wire lengths. Obviously, delay is very much dependent on the longest path length, which is no longer a global characteristic of all the wires together. Also, a lot of routing problems are very dependent on individual wires (such as antenna effects, technology and mask related distance rules, etc.). Hence, many researchers have tried to extend wire length models to individual wires, in my opinion without much success. The reason is that Rentian interconnect analysis inherently is a global thing and cannot be related to individual wire features very well.

Q2: What do you see as 3-5 key directions that the SLIP community and SLIP workshop will take in the next 10 years (and, why)?

Patrick Groeneveld.

I can suggest the following directions.

(1) Correlation accuracy is key. Find better ways to empirically validate the correlation accuracy of a system level interconnect prediction. How else would be be able to compare two predictions?

(2) Correlation accuracy is key, again. Modeling based on Rent's rule abstractions miss too many of the intricacies of an actual layout. There are discontinuities around the blocks, where channel capacities need to be taken into account. Power, clock and pad infrastructure can dramatically influence the available routing resources. Any model that does not take that into account will be useless for practical applications.

(3) Correlation accuracy is key (again!!). Crosstalk delay seriously affects the timing of actual wires on the chip. Minor changes in the neighboring wires can cause wire delay to vary by 2x in 28nm technologies. This seriously deteriorating the accuracy with which delay can be predicted at the system level. Correlation can be improved at the expense of cell area of routing resources (e.g., with spacing or shielding). SLIP tools could help in finding the sweet spot in this tradeoff curve.

(4) The non-committal 'Prediction' in 'SLIP' should morph into a more practical 'Routing' ('System Level Interconnect Routing' workshop!). Rather than produce pretty pictures or numbers, focus on producing wire topologies that have some chance of correlation with the final design. Such predictions are key in converging to a usable floor plan 2-D or 3-D. The floorplan problem is one of the big unsolved EDA problems, and its relevance will only increase in the next 10 years. Since any floorplan is only as good as its interconnect prediction, this is a key area to work on.

Lou Scheffer.

There is now a wide variety of interconnect resources, even on a single chip. Upper metal layers can be 8 or more times as big as lower layers. Some technologies, such as optical, offer fast times or higher bandwidth, but require large (and perhaps high latency) transmitters and receivers. Network on chip offers an entirely different set of tradeoffs. The problem of optimizing systems in a world of heterogeneous resources will only get worse with carbon nanotubes, graphene, RF interconnect, low voltage signaling, and other options. SLIP must grow to deal with these issues.

Similarly, the system level resources have grown. There is routing within packages, routing on the PC board, and routing on the back of the chip. 3D routing, with a few stacked chips, is now a relatively conventional technology. SLIP must learn to deal with all of these as well.

Coding technology, power consumption, system robustness, and SLIP will interact even more strongly. Correlations between signals are critical and must be handled better.

Biological systems, particularly nervous systems, have a host of SLIP-like issues. There are a number of different signaling mechanisms, such as synapses and neuromodulators, that have very different physical proximity requirements. The networks are intrinsically fully three dimensional, performance (and perhaps function) depend critically on the physical design, and complex tapering patterns are typical. Networks must be grown, not constructed de novo, and remain in operation as they are modified. Learning, in particular, is suspected to involve a host of SLIP-like issues, and must consider not only the current set of connections, but those that may be wanted in the future, or those that are no longer needed. First we need to understand these systems, and then at some very future date, design them.

Dirk Stroobandt.

In my opinion, the quest for individual wire features based on global interconnect models is a dead end. Progress in this area will only be possible with other than Rentian interconnect models. However, I do see a bright future for global interconnect models with the increase of abstraction levels in ESL. Architectural exploration at the system level still deals with a small amount of objects (parts of code, hardware building blocks). But that will soon change as systems get more and more complex. When such systems move to more complex systems with a lot more subsystems, the global nature will again dominate and Rentian models can gain momentum again. Rentian models will then allow for very fast estimates for ESL system level architecture exploration and in Networks-on-Chips research for large multicore networks. So, my belief is that SLIP will actually really move into the system level, hence the "S" in SLIP will be dominating.