

Modelling hardware using virtual platforms is becoming mainstream – so now attention is turning to the speed of simulation.

By **Chris Edwards**

Forget breadboards, the high-speed desktop PC is becoming the weapon of choice in developing software for system-on-chip designs



IF PEOPLE QUEUING up to produce standards is any indication, system-level design has finally penetrated mainstream chip development. It is more than ten years since engineers at UK mainframe maker ICL tried to popularise the idea of modelling hardware at the level of transactions rather than logic levels. Now the technique is being embraced around the world with a number of companies trying to kick-start standardisation efforts around system-level modelling.

Europe was the first territory to pick up on the idea of trans-

action-level modelling. The approach mostly throws out the concept of time when simulating hardware: everything is reduced to the level of messages flying around the model. It is possible to produce high-level models that are accurate to the level of the clocks that drive most electronic circuits. But the advantage of transaction-level modelling is that a simulation can be very fast, making it feasible to run applications software on top of the virtual hardware and not just boot code or small test routines. A number of companies have found that

using these prototypes lops months off design time by making it possible to have hardware developed in parallel with software.

The big hold-out on transaction-level modelling has been the US, but even there things are changing. Companies such as Intel and Texas Instruments recently joined the Open SystemC Initiative (OSCI), the organisation responsible for the language most commonly used in building virtual prototypes. "The situation is changing," says Limor Fix, general chair for the upcoming Design

Automation Conference (DAC) and associate director of Intel Research in Pittsburgh. "Reorganisations are happening inside companies to understand the challenge."

Bill Neifert, chief technology officer of model creator Carbon, says the move towards system-level design is one that the larger companies in the US are making: "I will know that it has hit the mainstream when a startup says they want to adopt it. Startups are more focused on getting one system or chip out of the door. Rather than putting a process in place. We will see ESL hitting



'The new chair of the working group seemed hell-bent on making sure that TLM 2.0 was released in the DAC time frame'

Mike Meredith

the mainstream when people say that is the best way to go."

The language that most people use today to create system-level models is SystemC. It is basically an extension of the C++ software language with additional class libraries to express concurrency and hardware-oriented concepts such as buses. However, in the US, the use of system-level languages may evolve differently to how things have progressed in Europe and the Far East.

Fix says the jury is still out on which language will be used for system-level design in the US. Intel was one of the companies that pushed for adoption of SystemVerilog, the language created by Codesign Automation, a startup that was bought by Synopsys. Fix reckons there will be competition between SystemC and SystemVerilog, particularly in the US, as to what engineers will use for transaction-level models.

"It will be quite a while before there is any convergence. But there is a good thing in competition: the customer always wins," she claims.

Although SystemC has a head start in terms of system-level modelling, it is not without its problems. The biggest hurdle is getting models from different companies – and even different departments – to work together. The Transaction-Level Modelling (TLM) specification created by the Open SystemC Initiative (OSCI) was meant to fix the interoperability problem.

The initial standard provided a great deal of freedom that engineers could use to express the way that interconnects such as buses could have models. In many cases, models from different companies intended to interface to the same bus could not be used together without creating wrappers to map transactions from one modelling style to another. The result was a mountain of models separated by a common language. So, OSCI set about creating a second version that would fix the problem.

Since work started a couple of

years ago, the TLM 2.0 standard has run into delays. "On the whole, the goals of the TLM 2.0 effort were pretty ambitious," argues Mike Meredith, president of OSCI and vice president of technical marketing at Forte Design Design.

Meredith said at the recent Design Automation and Test in Europe (DATE) conference that the standard is close, based on the outcome of a meeting held that week: "The new chair of the working group seemed hell-bent on making sure that it was released in the DAC time frame," he said.

The way that TLM 2.0 is constructed will not solve the interoperability probability at a stroke. But a number of companies are working towards updating their models to TLM 2.0 compatibility with other users expecting to use custom wrappers to connect their internally developed models to those created by vendors. Work by the Spirit Consortium may make the management of those wrappers a lot easier, with the result that people working in the area believe that TLM 2.0 will be widely adopted.

#### WIDE SUPPORT

Ian Mackintosh, president of interconnect standards group OCP-IP, says the members are backing TLM 2.0: "From the outset, a number of key members have been collaborating with OSCI. We were very much involved in the definition of TLM 2.0. We have shipped thousands of TLM kits overall. Now, we have embarked on a programme to move to the next generation of TLM kits."

A common concern about modelling using SystemC and TLM is performance. Throwing away timing helps greatly with simulation speed, but you are still faced with the issue of trying to model a complex SoC in software on a workstation: there are bound to be bottlenecks.

This is an area where Imperas, a company set up by some of the founders of SystemVerilog creator Codesign Automation, aims to make its mark: by providing more streamlined

models aimed at embedded software developers. The company decided that it would release some of its technology in open-source form as part of its Open Virtual Platforms (OVP) initiative. The aim of OVP is to provide fast models of processors in a multicore environment – as many SoCs now involve at least several on-chip processors – with the potential to link to SystemC models.

Imperas claims that it can hit speeds of hundreds of millions of instructions per second (MIPS) with simulations using the core OVP simulator on a typical development PC or workstation. To help with performance when synchronising tasks between processors, the linking hardware can be modelled in the OVP environment instead of SystemC or, for low-level detail, a hardware-description language.

A number of companies welcomed the initiative with some signing up as 'supporters'. However, companies such as Carbon Design Systems and Tensilica agreed to support the Imperas move on principle rather than as a result of seeing what Imperas would deliver. When OVP was launched as an open-source organisation, there was little actual source code available on the organisation's website.

Chris Jones, director of strategic alliances at Tensilica, tells *E&T*: "The creation of OVP by Imperas is a potentially disruptive event in the virtual-platform market. Consequently, it deserves our attention as we must always be sensitive to which design flows and methodologies our customers choose to employ.

"So far, there is little information on the simulator or APIs for models on the OVP download area, so assessing them for their usefulness and completeness is not possible at this time. We will continue to monitor their progress, and, provided there is sufficient customer demand, work with them to provide models of Tensilica processors for this platform."

Neifert says the engineers at Carbon have not had a chance to inspect the OVP programming interfaces but noted the importance of platforms to plug models into. Supporting another alongside TLM 2.0 and other environments would not be hard. "Talking at a conceptual level, we were able to endorse OVP because I know we will be able to integrate. We can easily talk to a programming interface."

#### SPEED HACKS

Synopsys was not among the list of supporters collated by Imperas but Schirrmeister said: "We really do welcome in this area activities in general to get software developers involved. Software developers are important. Everybody in the ecosystem has a role to play. If customers request that our models work in OVP, it is not rocket science to build a wrapper."

However, Schirrmeister questions the performance advantage of OVP over what is possible with TLM 2.0. "People are reporting multicore platforms running at 250MIPS with SystemC," he claims, noting that performance was an area that modelling pioneers such as Axis, which was bought by ARM, Coware and Virtio, later absorbed into Synopsys, worked on. Independently, the companies developed streamlined methods to reduce the amount of communication that the processor model needs with the SystemC hardware model.

Effectively, the processor simulations 'free wheel' until they absolutely need to synchronise with hardware. Memory accesses are not modelled using SystemC code but use 'backdoor' accesses. "In TLM 2.0, we now have all these elements. You can run ten processors and only need to synchronise them after 10,000 instructions or so," said Schirrmeister. The result is much better performance for software developers than if they tried to use a model based solely on SystemC.

Mark Burton, founder of GreenSocs and who is working ▶

## 'It's certainly the case that SystemC hardware models are not necessarily fast, but that's not the fault of SystemC'

◀ to set up a virtual-prototyping group within the Eclipse open-source development environment, agreed: "It's certainly the case that SystemC hardware models are not necessarily fast, but that's not the fault of SystemC. The modelling engineer can model how they like: they have the full power of C++. Imperas's model is also written using C++, I'm sure.

"The question is what the modelling engineer chooses to model, not the language. To take advantage of Imperas's technology, or to take advantage of SystemC, or to make use of ESL and the ESL design flow, modelling engineers need to move away from the detail, and towards modelling the intent of functional components."

### DETAILS TRADEOFFS

The question is: how much detail is appropriate? Imperas' approach of trying to model as much of the multicore infrastructure as possible at a high level should achieve high speed. But it might not have the accuracy required.

Limor Fix says multicore development is complicated by low-level interactions that will be invisible to software debuggers. This, she reckons, is likely to lead to the implementation of hardware monitors on the chips themselves that are able to log individual accesses to caches and shared memory at a high level of granularity. If the real hardware needs to be able to monitor interprocessor communication on a cycle-accurate basis, can you leave that out of an effective simulation before the actual hardware is ready?

Neifert says: "A lot of people write a lot of software that does not need a lot of accuracy. The details don't matter because your software doesn't rely on them. But, sometimes I need more accuracy than even TLM can support. We have found cases where we plugged RTL-accurate Carbon models in and the thing stopped working because the programmer's

untimed view didn't reflect actual behaviour.

"I have seen some high-level software have more hardware dependence than the programmers would like to believe. I definitely think you can write 100 per cent of your software on an abstraction that has no hardware model. But how much that will apply will vary on a use-case basis. Even on a user basis. And, at the end of the day, people will always want that implementation level check in there."

For OVP to take off, it will need people to start demanding that models come in that form. "It assumes that an ecosystem will come," says Frank Schirrmeister, director of product marketing for virtual platforms at Synopsys.

Some models already exist for the OVP platform, principally for cores developed by MIPS Technologies. But other companies, such as ARM, have forged ahead with their own modelling technology that already plugs into the TLM environment without changes. "They won't abandon that," Schirrmeister adds.

The bigger issue, says Schirrmeister is that users want to use modelling technology to represent the integrated system, not just the processor-memory subsystem. "Customers say the problem they need to solve is the integration at the end. The integration phase is mayhem."

Neifert notes the issue of market power in being able to drive a project such as OVP forward, contrasting the situation with the Eclipse virtual-prototyping platform (VPP) proposal. "It will be interesting to see how far VPP goes and if it gets much traction. I think that if you look at the various open-source initiatives that get a lot of traction, they have always had a major company behind it," says Neifert.

"With OVP, you have the situation where maybe Imperas is not a major company right now. But you can't argue with

the lineage of its founders. VPP does not yet have a big champion out there to fund the marketing battles."

However, it is not an either-or situation between OVP and VPP. "They are complementary," Burton says. "I don't think OVP is trying to provide a design environment like Eclipse and the VPP project is essentially there to try and help things like OVP, or other ESL tools, to make use of Eclipse as a design environment."

In a part of the design environment that has become used to slow progress on standards – yet where individual design teams have achieved quick turnaround results – the emergence of a series of specifications with varying degrees of openness marks a change in how the industry as a whole approaches chip design. The question is now whether the newcomers can make inroads into the market. ■



Consumer-product designers want to use more modelling for faster products