# Keeping things **spinning**

Multicore processors offer power saving benefits, but not without demanding more from operating systems. By **Philip Ling**.

It was identified a few years ago, by extrapolation, that transistor junction temperatures in integrated circuits could get as hot as the surface of the sun; or rather they wouldn't, because they would melt first. The real problem was that the timeframe for that extrapolation was disturbingly close: something had to be done.

The advent of 130nm features and below put a lot of focus on power density; particularly at the transistor level. Techniques to reduce leakage current received special interest, because the rise in static leakage was outpacing the increase in the number of transistors integrated.

An example is dynamic body biasing, developed by Intel, which varies the threshold voltage dynamically. Threshold voltage is a double edged sword; a low threshold means a fast transistor, but a leaky one. By varying the threshold dynamically, the transistor can be fast when needed and slow – and so less leaky – when not. The application of transistors has also evolved. Replacing power vias with transistors, for instance, allows whole sections of a chip to be turned off when not in use, reducing leakage current to zero.

Beyond that, design techniques – including asynchronous circuits, clock gating and dynamic voltage/frequency scaling – have all been implemented in recent years. These techniques address switching power successfully – the power actually used when the transistors are changing state. But even the gains made here struggle to deliver the headroom needed to appease evermore power hungry processors.

The solution, ironically, is to add more processor cores. It may sound strange, but closer inspection shows it makes perfect sense: all the techniques mentioned hitherto were developed to reduce active and static power – so long as the integrated circuit operates within certain performance parameters. Go outside those parameters and the power penalty is exponential. The answer is to aggregate the performance of multiple processors


"Linux is SMP ready. In the past, VxWorks hasn't been used in this way, but the ground work is there."

John Fanelli, **Wind River**

operating within their power parameters, returning both high performance and low power. However, the challenge then becomes running multiple threads and tasks across multiple cores efficiently.

## Lower power, higher bar

Pretty much every processor vendor now has a multicore strategy and now software vendors are responding in kind. That doesn't mean systems with multiple processors are new; they aren't.

Parallel processing has been around for a long time, predominantly in the form of distributing the same task across a number of processors. The vision now encompasses: load sharing across multiple cores; distributing different tasks across multiple cores; multiple cores executing different but codependent tasks; and having dissimilar tasks running on multiple cores, but using the same resources.

Compounding this challenge is the concept of homogenous (the same architecture) and heterogeneous (different architecture) cores running on the same piece of silicon. It raises the bar for operating systems.

It's yet another diversion for OS vendors like Wind River and QNX – both of which have bowed to customer demand to adopt field levelling technologies such as Linux, Eclipse and Posix. They've taken demand in their stride and successfully turned it into a competitive edge.

The most recent open source development adopted by commercial OS vendors is Transparent Inter Process Communication (TIPC), developed by Ericsson for telecommunication clusters. After its release to the open source community, it was soon ported to Linux and is now widely used in processor clusters for telecommunication and networking applications. In this kind of application, TIPC is reported to be around 35% faster than TCP/IP.

TIPC goes a long way towards enabling multiprocessor systems, by providing a transport layer and message passing interface that enables processes or tasks running under different operating systems, and on different processors, to communicate.

In March this year, Wind River announced it was extending the TIPC open source protocol to support VxWorks and contributing the extensions back into the open source community. "This is great news. VxWorks' support for TIPC shows that Wind River is serious about supporting interoperability with Linux and other OSes," said Jon Maloy, senior researcher with Montreal based Ericsson Research Canada. "They clearly understand the need for a standard IPC mechanism for device software. Their contribution to the TIPC project has been invaluable and their success once again proves that making TIPC easily portable was a smart strategy."

In October, QNX announced its support for TIPC. "QNX support for TIPC comes at a critical time, when designers of ATCA and multicore based communications systems may use Linux for the control plane, but need an rtos like Neutrino to achieve extremely fast and predictable performance on the data plane," said Romain Saha, network segment manager at QNX. "With this TIPC solution, system designers can leverage the best of the Linux and QNX worlds, while enjoying seamless interoperability between the two."

Standardisation is great for the customer, but requires suppliers to be more creative when it comes to differentiating their OS. No wonder, then, that the prospect of multicore processing has demanded careful thought; it represents a return to examining the strengths of the operating system and how they apply to multicore processing. As such, it is a significant development that has to be handled strategically.

The most common and widely deployed form of multiprocessing at the moment is symmetrical multiprocessing (SMP). John Fanelli, Wind River's vp of product management and planning, commented at Wind River's recent Regional Developers' Conference in Cambridge: "Linux is SMP ready. In the past, VxWorks hasn't been used in this way, but the ground work is there." Fanelli added he is planning to announce a multicore solution in the Autumn of 2006.

QNX is on the cusp of announcing its plans to further support multicore processors. The natural extension to SMP is AMP (asymmetric multiprocessing) – which supports advanced load balancing. The different between the two is that AMP would typically be running an OS on each core; either the same OS or a different OS on each core, depending on the application's requirements. QNX believes this can lead to a sub optimal solution, as neither OS has full control over the system, which could lead to resource management issues. It calls the solution it is proposing 'bound multiprocessing' and believes it represents 'the best of both worlds'. Further details will be available in the new year.

It isn't only large communication networks – where TIPC is applied – that will need to adopt multicore processing; eventually, it will permeate all the way down to consumer appliances. This evolution will not only meet processing demands, but also the need to operate within ever tighter power budgets