A new wave of virtualisation tools is helping to simplify the task of migrating, debugging, and optimising software for multicore processors. Able to provide a system wide view of multicore behaviour, virtualisation allows developers to quickly diagnose the complex interactions that characterise multicore designs and to pinpoint bottlenecks.

These benefits vary based on how virtualisation is being applied. For software developers, the major benefit is to provide a more flexible and powerful development platform than actual hardware. For the end product, virtualisation provides the flexibility of mixing operating environments on a single hardware platform which leads to a variety of benefits, including improved security, higher availability and reduced size to weight and power ratios.

It’s worth pointing out that one of the major issues with the word ‘virtualisation’ is that it means different things to different people. It helps to separate the technology into distinct application areas — software development, server provisioning, security and other applications.

At the high end of technology, virtualisation provides a computing platform that acts or simulates the system being targeted. In the embedded software arena, there are two major — and very different — applications of virtualisation that can create confusion when trying to understand their benefits: the sub classification of virtual prototyping, which is used to improve the software development process; and the provision of a virtual platform, which helps to improve the actual run time flexibility of a system, in other words, an end product that incorporates virtualisation.

Hastening adoption
For embedded applications, virtualisation can hasten multicore adoption. Trends suggest that virtualisation use is being driven by time to market demands, production cost pressures, increased consolidation of functions and the need for more flexibility.

“Flexibility comes from time to market demands for the ability to start developing your application and run it before you have the hardware,” explained Green Hills’ cto David Kleidermacher. “The gap between when software development begins and when hardware is actually available is increasing and, nowadays, systems are becoming more complex, so the software element becomes proportionally larger and time consuming. You no longer need wait until the hardware is ready to start developing your software.

“Virtualisation is also gaining prominence because desktop pcs are becoming fast enough to support its use at a reasonable speed,” he continued. “For example, Green Hills has just released its Multi 5.0 virtual prototyping environment which allows the user to run a simulation of an ARM at 100MIPS on a pc. Suddenly, customers can simulate at close to the same speeds as a real embedded system.”

Realtime embedded OS technology specialist QNX claims virtualisation can act as an abstraction layer beneath the OS. The fact that multiple compute cores can be abstracted or hidden from the OS itself allows you to take an OS that is not ‘multicore capable’ and run a rtos on top of a virtualised collection of processors.
QNX product manager for multicore Kerry Johnson said: “I see great value in virtualisation as it allows you to run multiple OSs and consolidate them onto a single higher power processor.

Clearly, virtualisation presents a challenge that companies strive to implement well to achieve adequate performance, whilst retaining the flexibility of the customers’ hardware choice. “The challenge we face is that our customers’ hardware is constantly changing,” noted Kleidermacher. “In the embedded world, this problem is exacerbated because the level of accuracy the customer needs will vary. We have to constantly adapt as new hardware platforms are released.”

According to Kleidermacher, the next challenge is performance. “Green Hills’ Padded Cell secure virtualisation technology allows you to run Windows on top of Integrity real time OS,” he affirmed, “and you can’t tell that it is not running native speeds. Virtualisation opens up performance possibilities previously unheard of on standard PCs. Padded Cell makes it possible to run two copies of Windows seamlessly from a single hardware box which you can’t do with a regular PC.”

Johnson suggests that in the embedded world of industrial automation, the advent of virtualisation and the increase in dual core processor technology is driving a demand to consolidate different real time OS platforms onto one processor.

“The point to remember is that when you virtualise, you still need to be able to guarantee real time,” he explained. “Virtualising by running multiple OS isn’t the whole story – you need to consider the connected devices in the peripheral and ensure your real time OS has the required access. A virtualisation layer that allows you to run a rtos is more difficult than what you would find in server applications, where real time isn’t as important. Open source solutions don’t translate well to the embedded world, although we are starting to see certain vendors offer solutions for the real time aspect. QNX’ approach would be to partner those companies with mature solutions to maintain the real time requirements we need to work in conjunction with another general purpose OS.”

Industry use of virtualisation varies dramatically, depending on how it is applied. In the server market, it is used to improve the management, provisioning and availability of servers. Another area of interest is the use of virtualisation as a means to reach higher levels of security, whilst maintaining the ability to reuse the vast existing software application base.

“Although software defined radio has been driven by the defence industry, it has many commercial applications,” Kleidermacher (below) observed. “Customers are demanding a single hardware platform that can talk different ‘real’ languages. This provides more interoperability and flexibility by taking some of the hardware functions of the radio and putting them into the software. The radio can then change its configuration dynamically and handle different kinds of radio communications.

“One Green Hills customer was using Integrity OS, but wanted to reuse its original networking software on top of Linux,” he continued. “Instead of two computers, it had one PowerPC controlling the software defined radio system. Padded Cell enabled it to run Linux as an application on top of Integrity, which meant it could reuse the original software. It now has separation between the Linux being used for legacy reasons alongside critical applications controlling the radio itself — all running directly on top of Integrity and all on the same computer.”

The last word goes to Johnson. “It’s still early days for virtualisation in the embedded arena, but what I envisage is the promise of virtualising the processor itself to obtain the kind of consolidation that you just cannot do without. This has a big impact by making the best use of processors and keeping the BoM for large scale systems in control.”

---

**Figure 1: Secure partitioning for multicore**

<table>
<thead>
<tr>
<th>Secure partition</th>
<th>50%</th>
<th>10%</th>
<th>15%</th>
<th>25%</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

QNX Neutrino RTOS

multi-core CPU, shared I/O, memory

---

Kerry Johnson, QNX

New Electronics www.newelectronics.co.uk 12 June 2007