One size fits all?

As the ARM architecture becomes more popular, will one size of BSP fit all?

By ***Philip Ling***

The embedded software domain has always exhibited less homogeneity than the enterprise world; the array of processor architectures and operating systems puts the 'Wintel' platform firmly in the shade.

Choice, however, isn't always a good thing, particularly when it takes incrementally more time and effort to assess the commonalities and differences between architectures. In the enterprise domain, application developers have a certain level of freedom here, as they needn't worry too much about the target architecture. Arguably, a similar trend is emerging in the embedded domain, as the ARM architecture continues its penetration.

Unlike the x86, ARM sports a range of architectures which, although largely binary compatible, are intended to address differing embedded systems needs. Is it possible that the dominance of ARM can result in a reduction – or at least slow down – in the rising level of complexity when developing embedded software applications?

For some time, the embedded industry has groomed the board support package (BSP) in order to provide greater accessibility to new platforms. Because they are closely linked to the operating system, the processor architecture and its integrated and external peripherals, BSPs are more numerous than their constituent parts. One important element of this equation is the processor, which arguably is likely to be ARM based.

As the term BSP has no real heredity, it can be applied liberally. Alan Harry, chief executive of embedded tools developer Crossware, pointed out: "It's clearly a software package that configures a particular microcontroller or microprocessor and possibly a particular board that uses that chip. Beyond that, it seems to be a name given to something that is otherwise vague."

A BSP will normally provide, at minimum, the device drivers needed to run a processor, while additional features may vary. IAR System's tools and applications manager Mike Skrtic explained: "In particular, device drivers are an absolutely essential component of a BSP; without drivers for the peripherals, the board could not be used to its full potential."

These views presuppose the processor forms the heart of a larger system, which will primarily be used for development purposes. But in isolation, device drivers are of minimal use. Skrtic added: "Typically, a BSP will include device drivers, demonstrations, example projects, peripheral stacks and debuggers. If they are included in an evaluation kit, such as IAR's KickStart, they may also include an RTOS awareness module and a limited demo version of an RTOS."

Adnane Senhaji, leader of Adeneo Software's European sales and marketing activities, accepts that device drivers may form the most important part of a BSP for software developers, but added: "Functionally speaking, the base of the entire BSP is the bootstrap code and the hardware abstraction layer (HAL); it initialises the platform and manages the cache and the interrupts."

Illustration: Don Seed

For BSPs supporting ARM architectures, therefore, the HAL forms an important and possibly bespoke element, Senhaji observed. "BSPs are an abstract layer between the OS and the hardware. In general, BSPs are organised in multiple layers. The first layer is dedicated to the processor architecture."

One size fits all?

While significant differences at the processor's architectural level remain, it seems it will always be necessary to implement an HAL, tying the BSP to the processor. The extent to which one HAL can accommodate many processors depends largely on how the processor core is implemented. Harry noted: "There is an issue of compatibility between chips from the same manufacturers. The on chip peripherals of Atmel's ARM chips are exceptionally compatible, so once a developer has mastered one chip, it is easy to move to another. Some manufacturers don't seem to understand compatibility; a developer has to start over when moving to another chip in the same family."

BSP providers face the same problem: if the on chip peripherals are compatible, then a BSP can be retargeted to another chip in the family; if not, the BSP has to be rewritten. Harry added: "We are in the process of introducing a feature that makes it easier to import the Atmel ARM example programs in to our IDE. An example is the lightweight TCP/IP stack Atmel has ported to its ARM chips."

While Senhaji believes continued diversity in peripherals will always require a commensurate diversity of drivers, he said some rationalisation could take place. Noting that ARM BSPs have three major components – the kernel, the drivers and boot code – he continued: "ARM cores are tending to have more common controllers. In the very near future, it's probable that ARM based chips will include multiple cores and some controllers will be aware of this, so the kernel part may become more generic."

With respect to the boot code, he said: "Several boot media are available. Depending on their constraints, such as cost, size, speed and reliability, different customers use different solutions. In this area, 'one size fits all' will not be possible until somebody develops a super fast, super reliable, large and cheap storage technology."

Because of the dependence on peripherals and, subsequently, their associated drivers, BSPs are generally developed for specific segments in vertical markets. However, applications in general are converging, making many features applicable across a number of areas.

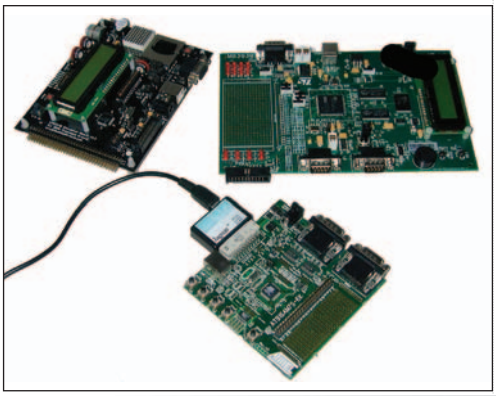
In response, there is a steady increase in the amount of third party IP developers may integrate in to their next application. What help can BSPs offer here?

Skrtic commented: "At the moment, BSPs don't do enough to enable developers to integrate third party IP, although a good start has been made and we're constantly striving to improve our compatibility with other vendors."

Senhaji confirmed: "Third party software IP and BSP suppliers should synchronise to make the application development easier for OEMs."

Despite ARM's penetration, it seems the embedded domain still exhibits a level of fragmentation that will extend the need and shape the evolution of BSPs. Given the BSP's importance, it is perhaps surprising that no special interest group or industry consortium is attempting to take control of this.

In Senhaji's opinion: "Developers wait for more abstraction between high and low levels, between applications and hardware. This will permit more code reusability and faster software porting from one hardware platform to another with minimum effort; that is, without having to rewrite drivers."



"Device drivers are absolutely essential; without [them], the board could not be used to its full potential." ***Mike Skrtic****, IAR Systems*

Skrtic noted: "What developers really need are real application examples, not just demos. This application specific quality is absolutely crucial to serving our customers better in the future."

Skrtic claims that IAR Systems is building competence centres in particular applications, such as automotive and medical, to develop specific application examples and to work with partners – including device vendors, OEMs and platform developers – to achieve this.

"A good example is a point of sale terminal, where we are working with both a platform developer and with an equipment manufacturer, who is happy to share code with us," Skrtic concluded. ■