Use case of MSP and PSP in Cortex M
Cortex M 中 MSP 和 PSP 的用例

techguyz 科技家伙 | over 11 years ago

Hi Experts, 您好，专家们，

What is the key difference between Master Stack pointer and the Process stack pointer and how an OS can take advantage of it ?

主堆栈指针和进程堆栈指针之间的主要区别是什么，作系统如何利用它?

Top replies 热门回复

Chris Shore over 11 years ago +2 ✓
Chris Shore +2 ✓

HI, Having two separate stack pointers allows the operating system to be safer and more robust. Usually, you would configure the operating system toy use Main Stack Pointer (MSP) and user applications...

嗨，拥有两个单独的堆栈指针可以使作系统更安全、更健壮。通常，您会使用 Main Stack Pointer （MSP） 和用户应用程序来配置作系统玩具......

Chris Shore 克里斯·肖尔 | over 11 years ago

HI, 你好

Having two separate stack pointers allows the operating system to be safer and more robust. Usually, you would configure the operating system toy use Main Stack Pointer (MSP) and user applications to use Process Stack Pointer (PSP). The switch from one stack to another then happens automatically when an exception is handled.

拥有两个单独的堆栈指针可以使作系统更安全、更健壮。通常，您将使用主堆栈指针 （MSP） 配置作系统，并将用户应用程序配置为使用进程堆栈指针 （PSP） 。然后，在处理异常时，会自动从一个堆栈切换到另一个堆栈。

The fact that the operating system and exception handlers use a different stack from the application means that the OS can protect its stack and prevent applications from accessing or corrupting it. You can also ensure that the OS does not run out of stack if the application consumes all the available PSP stack space - that means that there is always space on the stack to run an exception handler in the case of an error occurring.

作系统和异常处理程序使用与应用程序不同的堆栈这一事实意味着 OS 可以保护其堆栈并防止应用程序访问或损坏它。您还可以确保在应用程序消耗所有可用的 PSP 堆栈空间时，作系统不会耗尽堆栈 - 这意味着在发生错误时，堆栈上始终有空间来运行异常处理程序。

Note that you don't have to use both stack pointers. By default, the system will only use a single stack pointer (MSP) and must be manually configured to use PSP. Also, some Cortex-M microcontrollers do not support two stack pointers.

请注意，您不必同时使用两个堆栈指针。默认情况下，系统将仅使用单个堆栈指针 （MSP） ，并且必须手动配置为使用 PSP。此外，某些 Cortex-M 微控制器不支持两个堆栈指针。

Hope this helps. 希望这有帮助。
Chris 克里斯