

## Re: Preventing thread context switching

**Source:**

<http://www.tech-archive.net/Archive/WindowsCE/microsoft.public.windowsce.embedded.vc/2004-05/0481.html>

---

**From:** Ivan IV (*somebody\_at\_somewhere.spam.no*)

**Date:** 05/23/04

Date: Sun, 23 May 2004 12:30:07 +0200

"Offir Bakshitz" <offirb@adaptmedia.co.il> wrote in message  
news:%23U1gHiCQEHA.632@TK2MSFTNGP12.phx.gbl...

>>  
>> *Do you really need to prevent the context switch or you just do not want  
>> other threads to execute those instructions, too? If it is the latter  
> case,  
>> you may have a look at mutexes and semaphores. BTW, unless you are doing  
>> something \_really\_ low level, I cannot imagine a situation why you would  
>> need this. Not that such a situation does not exist, but in my  
>> multithreading past I never needed anything like that :-)*  
>>  
>> *Ivan*  
>>  
>  
> *I think I do. I am writting a driver for an IIC device. I am working with  
a  
> hardware port. Whenever context switch occurs, reading / writing to the  
> chips fails.*  
>

That's really low level. What about disabling interrupts, then.  
Context-switch is done usually on timer interrupt. It's quite heavy thing,  
but if you can minimize the time during which it is disabled, it may work.  
It's risky thing to do, because when you fail to enable it back, you end up  
with a dead system. I have no experiences with ARM assembler, but I guess  
there must be an instruction that prevents processor from executing  
interrupt routines. It used to be cli/sti back in 808x days.

Ivan