

Re: How To Avoids CLR GC operation

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.clr/2007-12/msg00014.html>

- *From:* "Chris Mullins [MVP - C#]" <cmullins@xxxxxxxxxx>
 - *Date:* Tue, 4 Dec 2007 09:51:31 -0800
-

You're up against a target I don't think you're going to hit based on advice from a newsgroup.

Go out, write some prototype code, and get it running. When it doesn't work, keep trying various approaches until you get something you like.

I suspect you're going to get the compact framework, running on a mobile device, to work as well as you're hoping. I would like to be wrong, but based on my work with the Compact Framework, you've got an uphill battle on your hands. I suspect GC will be one of your concerns, but will be far from the last...

—
Chris Mullins

"elwolv" <elwolv@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message <news:978EF835-0D8C-4DF9-93C9-7CBE46C4676C@xxxxxxxxxxxxxxxxxxxxx>

hello;

thank you for your response.

I am trying to use .net MF to issue motor control commands
commands can't be delayed for GC action.
Since GC is the system in .net MF it is the one who use/steals cycles from
the cpu.
By using static we can help continuous stream of commands without any
un-timed GC latency.

yes; .net MF/XP are not real time system and only windows CE/Linux are
real
time OS.,
However, .net MF CLR is the OS to be concern on the system.

my question now would be:

1- is there any specific technique for CLR (known to you as more
knowledgeable) to be used in this manner to help keep the commands flow

Re: How To Avoids CLR GC operation

without interruption?

2- short of using real time OS, I want to use .net MF; because it is easy to use and less time consuming and is really a good way to use embedded MCU. then what is the best ways to do that with the understanding, that we can't make .net MF a complete zero latency? there is enough RAM to hold a program.

thank you

--

elwolv

"Chris Mullins [MVP - C#]" wrote:

The question is easy to answer: Make the Static. This will put them on the high frequency heap, and