Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

System Construction Course 2019,

Assignment 4

Felix Friedrich, ETH Zürich

Introduction

Minos supports preemptive scheduling under certain preconditions that were discussed in the lectures. Background tasks execute in a round robin fashion in the background. Periodic tasks can preempt background tasks. The command scheduler of Minos is an example of a background task. We can protect ourselves from commands executing an infinite loop with a watchdog.

Lessons to Learn

- Learn to know the task scheduling mechanism of Minos.
- Understand and apply the mechanism of a watchdog.

Preparation

1. Update your repository
2. Open a console in directory assignments/assignment4
3. Make sure you have a proper kernel running on your RPI. If you need to recompile the kernel, you can do so by calling `oberon execute MakeMinos.txt`. You can then use module loading and are not required to link the kernel again for the rest of this exercise.

1 Watchdog and Tasks

Module Minos/RPI.Kernel.Mod contains procedures to setup the (largely undocumented) ARM watchdog registers `WDOG` (Watchdog) and `RSTC` (Reset Configuration). Bits 0 to 19 of the `WDOG` register provide a countdown register that, once it hits 0 will make the system reboot, provided register `RSTC` is set up accordingly. The frequency of the counter is 2^{16} Hz, thus a maximum of 16 seconds can be set for the watchdog countdown. Please refer to the implementation of `Kernel.StartWatchdog` in order to understand the semantics.

Once the countdown is activated, software must periodically update the `WDOG` register in order to prevent a reboot. In the case of a failed program, the watchdog is no longer updated which should result in a reboot of the system.

1. Complement procedure `Tasks.InstallWatchdog` to enable the watchdog and to install a *background* task that (periodically) resets the watchdog.
2. Test the watchdog by executing an infinite loop as a command.
3. What happens when you install the watchdog resetter as a *periodic* task? What does it actually test?

The starting point for this exercise is provided as module Tasks.Mod. It contains some example commands for installing periodic tasks.

Module Tasks can be compiled with the following command from the Oberon shell:

```
Compiler.Compile –p=Minos Tasks.Mod
```

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Documents

- System Construction Lecture 4 slides from the course-homepage
  `http://lec.inf.ethz.ch/syscon`