

ChibiOS-EmbrIO

Embryo Virtual Machines running on Chibi RTOS

Caccia Claudio

Matr. 751302, (claudiogiovanni.caccia@mail.polimi.it)

Report for the master course of Embedded Systems

Reviser: ing. Martino Migliavacca (martino.migliavacca@gmail.com)

Received: September, 16 2012

Abstract

The report describes the implementation of Embryo virtual machines on a Cortex-M3 ARM processor. Embryo is a library designed to interpret a subset of programs coded in a C-like syntax language known as *Small*. The library is tiny enough to be used on embedded systems with a reduced amount of memory. The idea behind this work is to make it possible to load, execute, unload and substitute different programs on an embedded system in a way that can be roughly compared to the one that we normally use in normal computers or in complex embedded systems like smartphones. This can make a small system highly flexible, by changing its behaviour at runtime, without the need to load the entire code and reboot. In order to achieve this goal, the Embryo library has been modified, in order to be used on an embedded system, and has then been integrated with a very performing RTOS like ChibiOS. Some programs have then been compiled, loaded and executed on a demoboard.

1 Introduction

When we think about Embedded Systems we can divide them into two main categories: on one side we have complex systems, like latest generation smartphones, where we can have a huge amount of memory, multicore processors, multiple interfaces and where we can download and use a lot of different applications and then cancel them without the need of switching the system off. On the other side we have small systems, like microcontrollers, which perform just one or few tasks for their entire life. We can just change some parameters only if we have implemented some callbacks on the system in order to update the values at runtime by using some communication interface (eg. Serial, Fieldbus...). We cannot change the overall behaviour of the system. For example, if we have implemented a PID controller, we can modify its parameters, but if we need to change the type of controller we need to code a new one, compile and then load it on the processor (eg. by means of a flash programmer or by using a bootloader).

In some environments it can be particularly interesting and time saving to change the behaviour of the system at runtime, eg. in robotics, in particular in cooperative robotics, it could be possible to change the way a single component acts without the need to keep all the possible configurations on board, or in artificial vision, it could be possible to change classifiers or algorithms and test them in a simple way.

So we investigated the possibility to implement a system based on virtual machines, so that one could load, execute and unload different components by selecting one or more program and launch it on a VM.

We needed a virtual machine small enough to be executed on an embedded system and the choice fell on *Embryo*. At the same time we needed all the features that a RTOS can provide (eg. task scheduling, IPC, memory management, HAL...) and we selected *ChibiOS*. Finally we needed an environment to implement and test the first basic features that we coded: for this purpose we selected a demoboard based on *ARM Cortex[™]-M3* processor.

In the following sections we describe in further detail these components.

1.1 Embryo

This is a citation [1] and here is another citation [2]. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed velit. Ut id orci ut quam placerat rhoncus. Pellentesque sed nisl. Aliquam sagittis nunc ut dui. Nullam elementum felis eget mauris! Curabitur nunc arcu, venenatis ac, commodo id, tristique molestie, urna? Donec in tortor vel augue rhoncus elementum. Nunc ac felis. Nam nulla quam, cursus sed, egestas nec, rutrum et, purus. Nullam sapien mi, ullamcorper rutrum, placerat eget, consectetur ut, metus. Integer porta, lacus non fermentum pretium, enim dui auctor ligula, ut faucibus lacus sem a metus. Maecenas posuere dui eu est. Ut massa urna, auctor quis, tempus et, sollici-

tudin quis, justo?

And this is the reference to a single column figure (see **Figure 1**). Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus dapibus convallis odio. Nunc sollicitudin laoreet ante! Vivamus dictum euismod orci.

Figure 1: Some single-column figure caption.

1.2 The second subsection of the first Section

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec et ligula. Nullam in libero. Donec dictum pede in justo. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam congue. Aliquam egestas. Nunc eu est ac nibh mattis vestibulum. Curabitur aliquet bibendum odio. Etiam hendrerit. Nunc a velit quis dui molestie consequat. Sed et turpis et mi feugiat tincidunt. Sed sollicitudin. Ut risus? Duis eget orci eu turpis consectetur fringilla? Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam tellus ligula, placerat vitae, tempor vitae, varius id; est! Nullam et ipsum eget tellus eleifend sollicitudin? Fusce urna massa, imperdiet vitae, convallis in; lacinia sed, tortor.

And this is the reference to a single column figure (see **Figure 2**). Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam consectetur neque at orci. Curabitur non metus. Praesent congue porta nisl. Suspendisse ultricies, sem ac ultrices aliquam, erat nisi fermentum est; a rhoncus mauris arcu eget nibh. Aliquam sollicitudin velit non erat. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer nulla diam, facilisis vel, accumsan sed; molestie egestas, massa. Fusce malesuada, ipsum et pulvinar aliquet, est dolor laoreet enim, quis porttitor erat mauris eget sapien. Integer vitae urna. Duis lectus.

2 The Second Section

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean magna. Nunc non ante eget nibh condimentum tempor. Nullam ullamcorper lectus eget mauris. Nam neque orci; rhoncus at, pulvinar quis, elementum sit amet, turpis. Mauris posuere nisi ut justo. Morbi non lorem vitae mauris interdum faucibus. Vestibulum ut sapien in augue faucibus fringilla. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Etiam vestibulum fringilla libero. Curabitur libero diam, hendrerit sit amet, ornare eget, imperdiet vel, purus!

2.1 The first subsection of the second Section

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nam consectetur ante at eros. Vestibulum mi nisi, venenatis sollicitudin, tempus sed, auctor id, tortor. Fusce orci. Duis tellus arcu, euismod sed, consequat sit amet, elementum vel, mauris. Curabitur leo diam; dapibus quis, condimentum vitae, dignissim ut, diam. Nulla et nulla eget elit volutpat sagittis.

2.2 The second subsection of the second Section

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris eget mauris. Nulla facilisi. Ut condimentum tempor eros? Integer metus mauris, consectetur sit amet, tempor a, facilisis eu, nisl. Vestibulum at turpis. Ut vitae tortor pretium nisl vestibulum blandit. Nulla nibh urna, semper et, elementum at, mattis ut, nisi! Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Morbi vel ligula eget lacus convallis venenatis. Aliquam lacinia tincidunt felis. Ut dui.

References

- [1] Ramsey, N.: Learn technical writing in two hours per week. Technical report, Harvard University (2006)
- [2] Hughes, S.P.J.L.J.: How to give a good research talk. SIGPLAN Notices (1993)

Figure 2: Some wide-figure caption.