

Under the hood: Bare Metal Embedded Programming in C

Wednesday, August 29, 2018 1:00 PM (5 hours)

A single plain C file is sufficient to express an embedded program.

As the Arm Cortex-M architecture is designed with C-Code in mind, no assembly level system bring up code is required. This workshop will teach you how to program C code on top of a bare metal CPU without an operating system or support libraries like libc.

It will give you insight on how linkers can be configured to run your program at the right location and placing data. We will use the free arm-gcc toolchain and related tools from the toolchain to analyze the program on assembly level to understand better how C-language is mapped to machine code depending on the chosen compiler optimization level and linker settings.

The workshop will further introduce you to how low level features like stacks and interrupts are used and how they map onto Arm Assembly code. One of the purposes of this course is to lay out the programming methods for talking to hardware in a minimal configuration. Our broader target is a better understanding of interaction with low level hardware and toolchains for embedded systems.

Last, but not least we will present debug techniques for low level / OS-development and might talk about security features of the used microcontroller platform.

In case you're interested in reading material on the topic, we recommend "The Definitive Guide to ARM® Cortex®-M3 and Cortex®-M4 Processors 3rd Edition" –but it will be by now means required for participating in this course.

An internet connected laptop is required for participating in this workshop - please install the latest version of Docker on your system and verify its running and that your system is updated. We will provide a Docker-based Linux environment for you with a pre-installed arm-gcc toolchain.

Basic knowledge of the C-programming language is required.

Summary

Presenter: MERIAC, Milosch (Bitmanufactory LTD)

Session Classification: Tutorials