

(Embedded) Linux for Professionals

Day 1

- ☐ **Scheduling in Linux**
- ☐ **Interprocess-/Interprocessor communication**
Posix IPC (Shared Memory, Semaphores, Signals, Sockets (MessageQueues), Pipes, Dbus
CORBA
- ☐ **„Best Practice“ recommendations for application development**
Do's and don'ts
Analysis of applications and structured layering of applications
- ☐ **GUIs (graphical user interfaces) – X; short intro to Qt and E17**
- ☐ **Kernel – roll your own (ryo) Linux kernel**
Cross development for ARM architecture
- ☐ **Driver development**
Kernel build-system / integration of your own device driver
Driver model (structure) of Linux
Driver details – shown on the character device driver

Day 2

- ☐ **Real-time and Linux**
Presentation of different approaches (RTAI, Xenomai, Preempt-RT)
Design principles and structure of Preempt-RT patch
Test programs for user- and kernel space
Real-time application development (process, thread, driver, ..)
„Best Practice“ recommendations
Do's and don'ts
Analysis of applications and structured layering of applications
- ☐ **Measurements (function generator, oscilloscope required)**
external trigger signal will be analysed
Analysis of real-time challenges (latencies, jitter, runtime jitter, ..)
Tools for analysis like cyclicttest, floodping, ..
Presentation and use of graphical tools (if available)

Day 3

- 📦 **Error analysis / debugging of Linux with build-in tools**
use of ftrace - debugging, tracing,
Performance analysis; appropriate tools and methods
- 📦 **Introduction to packet management systems for Linux, shown on Debian**
Principles and idea of a packet management system
Structural design of Debian as an example
- 📦 Reasonable HW infrastructure for Linux development
- 📦 Reasonable SW Infrastructure for Linux development (Build System, version control system, distributed development ...)

Requirement:

Personal notebook with CPU at least Intel Dual-Core or similar, at least 2 GByte RAM, 15 GByte free HDD, VMware player or Host V3.0. already installed.

VMware Image with required tools and examples will be delivered respectively will become provided in advance to the training.

Please take care for a not limited access to the internet in order to be able to download required additional software.

Software:

Linutronix provides an USB HDD with an x86 64-bit based Debian system for the host system, a Debian and a Codesourcery toolchain and for the target system an ARM Linux (running in a virtual machine). The HDD is a gift for the participant and can be taken home for further studies.

Hands-On training:

Scheduled are hands-on examples for the following themes: shell, cross compiling, and cross debugging (hardware could be provided by linutronix; please contact us if wished).

Hands-on examples might be limited to 2 persons for a single device

Number of participants:

Due to our experience we know that a single instructor could coach a maximum of 6 persons. Our courses are therefore limited to this number of individuals.