

## Workshop

# Compact Embedded Linux

Beim Einsatz von FPGA-Designs im Zusammenspiel mit einem prozessorbasierten System bietet sich Linux als offenes Betriebssystem an. Doch wie erstelle ich ein Embedded-Linux System für meine Hardware-Plattform wie z.B. für den XILINX ZYNQ SoC? Von der Entwicklungsumgebung über den Boot-loader und dem Linux Kernel mit Root-Filesystem werden alle Komponenten eines Embedded-Linux-Systems diskutiert, angepasst und verwendet. Dabei kommen nur freie Werkzeuge wie GNU Compiler, u-boot oder BusyBox zum Einsatz. Hierdurch erlernt der Teilnehmer nicht nur wie ein Embedded-Linux-System erstellt wird, sondern er ist auch in der Lage das Linux System für sein eigenes Projekt anzupassen.

Dieser Kurs ist praktisch orientiert, so dass Abschnitte durch Übungen gefestigt werden. Die verwendete Zielplattform ist der XILINX ZYNQ, das erlernte Wissen lässt sich jedoch auch auf andere Technologien wie z.B. Prozessorsystemen anwenden.

### Anwendbare Technologien

Prozessor basierende Embedded-Linux Systeme wie XILINX ZYNQ und andere.

### Voraussetzungen

Erfahrungen mit dem Umgang mit Linux als Betriebssystem sind wünschenswert, jedoch nicht notwendig. Es werden jedoch in diesem Training die wesentlichen Linux Grundlagen für Hardware Entwickler diskutiert. Des Weiteren sollten die Teilnehmer grundlegendes Wissen über Makefiles und C-Programmierung mitbringen.

### Dauer und Kosten

3 Tage, € 2.100,- netto pro Teilnehmer inklusive ausführlichen Schulungsunterlagen sowie Pausengetränken und Mittagessen

## Agenda

### Linux Basics for Hardware developers

- Linux Root filesystem concept
- Using the Console
- Scheduler, Processes and tasks
- Cross compiler and libraries
- Patching

### Boot loader

- General boot process
- U-boot configuration and compilation

- Hardware debugging with u-boot
- Network boot for practical development

### Linux kernel and root file system

- Linux kernel architectural basics
- Boot process
- Flatten Device Tree support
- Linux folder hierarchy
- File systems and wear leveling
- C library glibc, eglibc, uclibc