Embedded Design with PetaLinux SDK

Embedded Software 4

EMBD22000-13-ILT (v1.0)

Course Specification

Course Description

This intermediate-level, two-day course provides embedded systems developers with experience in creating an embedded PetaLinux SDK operating system on a Xilinx MicroBlaze™ processor development board. The course offers students hands-on experience on building the environment and booting the system using a basic, single-processor System on Chip (SoC) design with PetaLinux SDK on the MicroBlaze processor.

This course also introduces embedded Linux components, use of open-source components, environment configurations, network components, and debugging/profiling options for embedded Linux platforms. The primary focus is on embedded Linux development in conjunction with the Xilinx tool flow.

Level – Embedded Software 4
Price – $1200 or 12 Xilinx Training Credits
Course Duration – 2 days
Course Part Number – EMBD22000-13-ILT
Who Should Attend? – Embedded software developers interested in customizing the PetaLinux kernel on a MicroBlaze processor design for a Xilinx embedded processor system
Prerequisites

- *Essentials of FPGA Design* (introductory FPGA design course)
- *Embedded Systems Software Development* course (software development for FPGA embedded systems course)

Software Tools

- Xilinx ISE® Design Suite: Embedded or System Edition 13.2
- PetaLinux v2.1

Hardware

- Architecture: Spartan®-6 FPGA*
- Demo board: Spartan-6 FPGA SP605 board + FMC I/O card

* This course focuses on the Spartan-6 architecture. Check with your local Authorized Training Provider for the specifics of the in-class lab board or other customizations.

After completing this comprehensive training, you will have the necessary skills to:

- Explain what an embedded Linux kernel and device driver architecture requires
- Use the hardware interfacing options for the MicroBlaze processor
- Create a working MicroBlaze processor Linux system using the Xilinx Embedded Developers Kit (EDK) and the PetaLinux Software Development Kit (SDK)
- Build custom hardware cores and device drivers

Course Outline

Day 1

- Embedded Linux Overview
- **Lab 1.1:** A First Look
- Introduction to PetaLinux SDK
- **Lab 1.2:** Build and Boot an Image
- Application Development and Debugging
- **Lab 1.3:** Application Development and Debug
- Networking and TCP/IP
- **Lab 1.4:** Networking and TCP/IP
- Device Drivers, User Space I/O, and Loadable Modules
- **Lab 5:** Accessing Hardware Devices from User Space

Day 2

- Board Bring Up with PetaLinux SDK and Xilinx Platform Studio
- **Lab 2.1:** Base System Builder and Board Bring Up
- Custom Hardware Development and Interfacing
- **Lab 2.2:** Custom Hardware Development
- Device Driver for the Hardware
- **Lab 2.3:** Custom Driver Development

Lab Descriptions

- **Lab 1.1: A First Look** – Log in to the MicroBlaze processor Linux system and make comparisons between the embedded Linux and desktop Linux environments.
- **Lab 1.2: Build and Boot an Image** – Explore the Linux configuration menus, and build the MicroBlaze processor Linux kernel and applications. Download the resulting system image to the development board.
- **Lab 1.3: Application Development and Debug** – Create a simple user application with PetaLinux tools and debug the application with GDB.
- **Lab 1.4: Networking and TCP/IP** – Explore the kernel configuration menu and enable Linux TCP/IP networking. Log in to the MicroBlaze processor Linux system by using telnet. Transfer files to and from Linux by using FTP. Use the Network File System (NFS) to mount your host file system. Build and experiment with web-based applications under Linux.
- **Lab 1.5: Accessing Hardware Devices from User Space** Access a hardware device directly from user space. Use the UIO framework to access a hardware device. Experience loading and unloading kernel modules.
- **Lab 2.1: Base System Builder and Board Bring Up** – Use BSB and EDK to create a simple Linux-capable design. Use PetaLinux to create a new embedded Linux target for the hardware platform.
- **Lab 2.2: Custom Hardware Development** – Design a customized IP core. Integrate an IP core with the AXI or FSL interface and debug.
- **Lab 2.3: Custom Driver Development**– Write a UIO program to access the PWM AXI IP core. Or, use petalinux-new-module to create a new kernel module. Write an application to test the FSL PWM controller with the PetaLinux generic FSL driver.

Register Today

To register for this course or to see a list of currently scheduled classes, please visit our secure **Online Store**.

To request a public or private class, inquire about course offerings, or any other specific Xilinx training needs, please contact Faster Technology through one of the following:

Web: **Request a Class**
Email: **registrar@fastertechnology.com**
Phone: 281-391-5482

Visit **www.FasterTechnology.com/training-courses** to see our full line of Xilinx education courses in the areas of FPGA Design, Embedded Systems Development, Connectivity, DSP Design, Languages, and CPLD Design.