

Course Description

This course provides embedded systems developers experience with creating an embedded Linux system targeting Xilinx SoCs using the PetaLinux tools.

The course provides experience with:

- Using open-source embedded Linux components
- Using the PetaLinux tool design flow
- Creating and debugging an application
- Building the environment and booting the system using the Arm® processors available in Xilinx SoCs
- Customizing the root file system
- Configuring the Linux environment and network components
- Developing custom hardware and custom drivers

The primary focus is on embedded Linux development in conjunction with the Xilinx tool flow.

What's New for 2021.2

- PetaLinux Booting and Packaging module: Added information on booting a PetaLinux image on hardware with QSPI or OSPI
- PetaLinux Application Debugging module: Added more details on debugging a Linux kernel
- All labs have been updated to the latest software versions

Level – Embedded Software 4

Course Details

- 2 days live instructor led training (in person or online)
 - 17 lectures
 - 9 labs

Price – \$1,600 or 16 Xilinx Training Credits

Course Duration – 2 days

Course Part Number – EMBD-PLNX

Who Should Attend? – Embedded software developers interested in customizing a kernel using PetaLinux on the Arm processors available in Xilinx SoCs

Prerequisites

- [Designing FPGAs Using the Vivado Design Suite 1](#)
 - Introduction to FPGA Design
- [Designing FPGAs Using the Vivado Design Suite 2](#)
 - Designing with the IP Integrator
 - Creating and Packaging Custom IP
- [Embedded Systems Software Development](#)
 - Software development for embedded systems course

Alternative (and better) course for experienced engineers who can absorb a lot of information quickly:

- [Xilinx Rapid Development Embedded Design](#)

Software Tools

- PetaLinux Tools 2021.2
- Vivado® Design Suite 2021.2
- Vitis™ unified software platform 2021.2

Hardware

- Architecture: Zynq® UltraScale+™ MPSoC
- Demo board: Zynq UltraScale+ MPSoC ZCU104 board or Versal® AI Core Series VCK190 Evaluation Kit

* This course focuses on the Zynq UltraScale+ MPSoC. Check with [Morgan Advanced Programmable Systems, Inc.](#) 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 is
- Create a PetaLinux project to configure and build an image
- Create a working Arm processor-based Linux system using the Vivado Design Suite and PetaLinux tools
- List various hardware interfacing options available for the Arm processor
- Describe the Linux device driver architecture
- Build custom hardware cores and device drivers using the user space I/O (UIO) framework

Course Outline

Day 1

- Introduction to Embedded Linux**

Introduces embedded Linux, including a brief architectural overview, as well as some of the reasons for its rising popularity as an embedded OS. Also introduces the concept of toolchains and cross-compilation. {Lecture}
- Embedded Linux Components**

Describes the various components required for embedded Linux platforms (including the kernel image, root file system, and boot loaders) and how the components affect the booting of Linux on these platforms. {Lecture, Lab}
- Driving the PetaLinux Tool**

Covers the functionality, inputs, and outputs of the PetaLinux tools as well as the project directory structure generated by the PetaLinux tools. Basic PetaLinux commands are also introduced. {Lecture, Lab}
- PetaLinux Tool Design Flow**

Provides a brief description of the PetaLinux tool design flow and describes in detail various PetaLinux commands (including petalinux-create, petalinux-config, petalinux-build, petalinux-package, and petalinux-boot) and their example use cases. {Lecture}
- PetaLinux Application Development**

Introduces core concepts for developing, customizing, and running software applications in an embedded Linux environment. {Lecture, Lab}
- Customizing the Project**

Analyzes different configuration options provided by the PetaLinux tool for firmware version, rootfs type, boot image storage, and primary flash partition. Also describes external file system boot configuration. {Lecture}
- Customizing the Root File System**

Provides a brief description on customizing the rootfs for embedded Linux components such as libraries, applications, modules, layers, recipes, and packages. {Lecture}
- Networking and TCP/IP**

Discusses how the TCP/IP networking stack can be used to improve productivity during embedded product development by supporting network data communication, network control/status management, and firmware and hardware upgrades. {Lecture, Lab}
- PetaLinux Booting and Packaging**

EMBD-PLNX (v1.0)

Course Specification

Describes how to package and then boot a PetaLinux image via QEMU, SD card, JTAG, and TFTP. {Lecture}

Day 2

▪ PetaLinux Application Debugging

Describes how to debug software applications running on an Arm processor using the system debugger (TCF agent) or GNU debugger (GDB). {Lecture, Lab}

▪ Upgrading the Workspace

Describes the petalinux-upgrade command and how to upgrade PetaLinux project software components without changing the host tool components. {Lecture}

▪ Basic Hardware Design Process with the Vivado Design Suite

Describes the complete board bring-up process, which includes the hardware design as well as Linux image creation for the hardware. {Lecture, Lab}

▪ Linux Device Drivers Overview

Provides a brief overview of Linux device drivers and their requirements. Also describes what a device tree is and how it is generated. {Lecture}

▪ User Space I/O and Loadable Kernel Modules

Introduces two lightweight approaches for accessing the physical memory of devices from user space: direct access through the `dev/mem` virtual device and the user space I/O framework. Also covers the role and usage of loadable kernel modules. {Lecture, Lab}

▪ Custom Hardware Development

Describes the Create and Package IP Wizard and how it can be used to create a variety of architectural options for interfacing a system with custom processing hardware. {Lecture, Lab}

▪ Custom Driver Development

Discusses device driver options to match custom hardware devices and how to use the provided interfaces to read and write to the devices. {Lecture, Lab}

▪ PetaLinux: Advanced Configurations

Reviews how to modify advanced configuration settings using the PetaLinux tool. These configurations include selecting the Linux components for the build, enabling automatic configuration for a selected component, customizing how the Linux system interacts with the underlying hardware platform. {Lecture}

Register Today

Morgan Advanced Programmable Systems, Inc. (Morgan A.P.S.) delivers public and private courses in locations throughout the central US region; including Iowa, Illinois, Kansas, Minnesota, Missouri, Nebraska, North Dakota, South Dakota, and Wisconsin.

Visit morgan-aps.com/training, for full course schedule and training information.



You must have your tuition payment information available when you enroll. We accept credit cards (Visa, MasterCard, or American Express) as well as purchase orders and Xilinx training credits.

Student Cancellation Policy

- Student cancellations received more than 7 days before the first day of class are entitled to a 100% refund. Refunds will be processed within 14 days.

- Student cancellations received less than 7 days before the first day of class are entitled to a 100% credit toward a future class.
- Student cancellations must be sent [here](#).

Morgan A.P.S. Course Cancellation Policy

- We regret from time-to-time classes will need to be rescheduled or cancelled.
- In the event of cancellation, live on-line training may be offered as a substitute.
- Morgan A.P.S. may cancel a class up to 7 days before the scheduled start date of the class; all students will be entitled to a 100% refund.
- Under no circumstances is Morgan A.P.S. responsible or liable for travel, lodging or other incidental costs. Please be aware of this cancellation policy when making your arrangements.
- For additional information or to schedule a private class contact us [here](#).

Online training with real hardware

During the Covid-19 period, some companies do not allow their staff to participate in live in-person training.

- Consequently, Morgan Advanced Programmable Systems, Inc. has set up a training VPN where engineer participants can take classes online using the same computers and devCards used during in-person training.
- Even better, and upon request, you can use these computers after hours on training days to experiment with labs. This is not possible for in-person training.
- Additionally, just like in-person training, the laptops and devCards, tools, OS, and licensing are set up in advance.
- In some ways, live online-training is better than in-person...for example, you can grant the instructor permission to look at your Vivado, PetaLinux terminal, or Vitis for extended periods of time if your lab is not going exactly as planned to a missed step.
- This is often more comfortable than two engineers crowding around a laptop screen.

Taking remote training also allows you to learn some tips and tricks for working remote. Whether your devCard is in the lab down the hall, or across the world via VPN, you can control your Xilinx based device quickly and efficiently.