

Course Description

This course provides embedded systems developers with experience in creating an embedded Linux® system targeting AMD SoCs using the Yocto Project®.

The course provides experience with:

- Using open-source embedded Linux components
- Configuring the Linux environment and network components
- Utilizing a cross-reference guide for transitioning from PetaLinux to Yocto
- Generating Yocto machine configuration files using base hardware design artifacts
- Building Linux images using the Yocto design flow
- Booting the system using the Arm® processors available in AMD SoCs
- Reviewing BSPs and package groups and customizing it for AMD SoCs using Yocto design flow
- Mapping developer personas in the AMD Embedded Development Framework (EDF) to workflow roles
- Leveraging the Software Hardware Exchange Loop (SHEL) tools (SDTGen, Lopper, and gen-machine-conf) for Yocto machine configuration

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

What's New for [2025.1]

- Added new modules:
 - Using the AMD Embedded Development Framework (EDF)
 - Introduction to the Software Hardware Exchange Loop (SHEL) Flow
- All labs have been updated to the AMD EDF flow and latest software versions

Level – Embedded Software 4

Course Details

- 1 day instructor led training (online or in person)
- 9 lectures
- 3 labs

Price – \$800 or 8 AMD Training Credits

Course Part Number – EMBD-YOCTO

Who Should Attend? – Professionals who design and develop embedded systems and need to customize Linux for AMD SoC solutions

Prerequisites

- Familiarity with the Linux command line and basic Linux commands
- Basic understanding of embedded systems and their components
- Experience with version control systems (e.g., Git)
- Familiarity with AMD SoCs and evaluation boards is a plus

Related Course

- [Xilinx Rapid Development Embedded Design](#)

Software Tools

- Yocto Project

Hardware

- Zynq™ UltraScale+™ MPSoC ZCU104 board*
- Versal™ AI Edge Series VEK280 board*

* This course focuses on the Zynq UltraScale+ MPSoC and Versal adaptive SoC architectures. 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:

- Describe embedded Linux systems and their applications in AMD SoCs
- Describe the Yocto Project and its significance in building custom Linux distributions for embedded systems
- Explore the main components of the Yocto Project and their roles in the build process
- Utilize a cross-reference guide for transitioning from PetaLinux commands to Yocto commands, which are used for building and customizing Linux distributions on AMD SoCs
- Identify the role of board support packages (BSPs) and package groups in the Yocto Project, and how to create and customize them for AMD SoCs
- Boot an evaluation board using Yocto Native BSPs
- Describe the developer personas within the AMD EDF and their roles across the embedded development workflow
- Describe the SHEL flow and its key tools, which help in generating Yocto machine configuration files using base hardware design artifacts

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}
- **Introduction to the Yocto Project**
Introduces the Yocto Project and its significance in building custom Linux distributions for embedded systems. {Lecture}
- **Yocto Components**
Describes the various components required for building a Linux image in Yocto and how these components affect the building and booting of the image. {Lecture}
- **PetaLinux to Yocto: Command Cross-Reference**
Provides a cross-reference guide for transitioning from PetaLinux commands to Yocto commands. {Lecture}
- **Yocto Build Flow for AMD SoCs**
Discusses how to configure the Yocto build system for creating Linux images using the Yocto Project for AMD SoCs. {Lecture, Lab}
- **Introduction to Layers and Recipes**
Introduces the concept of recipes in Yocto and how they are used to manage software components. {Lecture}
- **BSPs and Package Groups in Yocto**
Describes the role of board support packages (BSPs) in the Yocto Project, including how to create and customize them for AMD SoCs. Also explores the concept of package groups and how AMD-specific package groups can be utilized and customized. {Lecture, Lab}
- **Using the Embedded Development Framework (EDF)**

EMBD-YOCTO (v1.0)

Course Specification

Discusses the AMD Embedded Development Framework (EDF) and its role in accelerating platform-level development. Also explains the developer personas within the EDF. {Lecture}

▪ **Introduction to Software Hardware Exchange Loop (SHEL) Flow**

Outlines the Software Hardware Exchange Loop (SHEL) Flow as part of the EDF and reviews the key tools used in the flow such as SDTGen, Lopper, and gen-machine-conf. {Lecture, Lab}

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 AMD 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 or in person training with real hardware

- 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 AMD based device quickly and efficiently.