Designing with the AMD Embedded Development Framework (EDF)

EMBD-EDF (v1.0)

Course Specification

Course Description

This course introduces the AMD Embedded Development Framework (EDF), a unified software framework for AMD adaptive SoC and FPGA platforms. The EDF provides a structured development methodology, pre-built software stacks, and integrated tools to accelerate Linux®-based embedded system development.

The emphasis of this course is on:

- Understanding the supported and recommended development flows for hardware and embedded software
- Identifying different design entry points and mapping role-based development within the EDF workflows
- Simplifying platform-level development by using pre-built embedded software stacks, pre-configured board support packages (BSPs), and open-source tools
- Leveraging the Software Hardware Exchange Loop (SHEL) flow and using tools like SDTGen, Lopper, and gen-machine-conf for hardware software integration
- Exploring the migration guide to use the EDF instead of PetaLinux

The primary focus is to gain hands-on experience with the EDF tools and workflows and learn how the EDF enables rapid prototyping and production-ready solutions through pre-built images.

Level – Embedded Software 4

Course Details

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

Price – $800 or 8 AMD Training Credits

Course Part Number – EMBD-EDF

Who Should Attend? – Embedded software developers and system designers working on AMD adaptive SoC or FPGA platforms who want to build Linux-based systems using the AMD Embedded Development Framework

Prerequisites

- Basic understanding of embedded systems and Linux-based development
- Familiarity with Yocto Project® concepts and workflows
- Awareness of hardware/software integration concepts in the AMD ecosystem
- Familiarity with AMD SoCs and evaluation boards is a plus

Software Tools

- Yocto Project
- Vivado™ Design Suite 2025.2
- Vitis Unified IDE 2025.2
- Hardware emulation environment:
  - VirtualBox
  - QEMU
  - Ubuntu® desktop

Hardware

- Zynq™ UltraScale+™ MPSoC ZCU104 board*
- Versal™ AI Core Series VCK190 board*
- Versal AI Edge Series VEK280 board*

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

- Build and customize Linux-based platforms using the AMD Embedded Development Framework (EDF)
- Develop and deploy applications using the EDF-provided SDKs, pre-built images, and workflows
- Migrate existing PetaLinux-based projects to the AMD Embedded Development Framework

Course Outline

Day 1

- **Overview of the Embedded Development Framework**
  Provides an overview of the AMD EDF and its role in embedded platform development. {Lecture}
- **EDF Software Stack and Role-Based Development Flow**
  Introduces the software stack, including Linux BSPs, pre-built disk images, tools, firmware, utilities, and reference designs and explains role-based development flows. {Lecture}
- **Application Development and Deployment Using the EDF**
  Introduces software and hardware application development flows and their deployment using the EDF Yocto workflows. {Lecture}
- **Linux OS Integration and Yocto Customization**
  Covers building and customizing the Linux OS using the Yocto Project for AMD SoCs. {Lecture, Lab}
- **Introduction to the 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}
- **Custom Hardware Development Using the EDF**
  Covers custom hardware workflows, including Segmented Configuration and dynamic PL reload. {Lecture, Lab}
- **Migration from PetaLinux to the Embedded Development Framework**
  Explains the migration from PetaLinux to the Yocto-based EDF flow and the mapping of common build and customization tasks. {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 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 has 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.