

Introduction to Embedded Linux Training Course Outline

Course Description:

Do you want to learn how to build embedded Linux systems? Looking for a good embedded Linux primer? Our Introduction to Embedded Linux training course describes how to apply Linux, the free, open source operating system, to embedded computing devices. Topics include: a brief introduction to Linux and the open source philosophy, installing a cross-development environment, using the Eclipse IDE, running and debugging applications on an embedded target, configuring and building the Linux kernel, booting the target, and network applications. Using this embedded Linux training course you will soon be on your way to developing embedded Linux systems.

Course Objective:

As a result of taking this Introduction to Embedded Linux online training course, you will understand how to:

- Set up a cross development environment on a Linux workstation
- Create, build, and debug applications on a target board using the Eclipse IDE
- Write multi-threaded and networked applications
- Configure and build the Linux kernel
- Install and boot the application in product form

Course Prerequisites

Attendees should be very comfortable programming in C. Some familiarity with Linux is helpful, but not essential.

As this is very much a hands-on class, you'll get more out of the class if you have the target hardware on which the examples are run. This would be the Embedded Linux Learning Kit from Intellimetrix, which is offered at a reduced price to GogoTraining members. Go to www.intellimetrix.us/gogogh.

Modular Outline:

Module 00: Course Introduction

Module 01: Introducing Linux

- What is Open Source?
- Installing Linux
- Linux features
- Tour of the file system
- The Shell

Module 02: KDE Graphical Desktop and the Command Shell

- Elements of KDE

- File Manager
- K Menu
- The Shell
- Installing course software

Module 03: The Eclipse Development Environment Part 1

- Introduction and background
- Workbench elements
- Eclipse menus

Module 04: The Eclipse Development Environment Part 2

- Creating a project – the C/C++ perspective
- Debugging with Eclipse
- Configuring Eclipse

Module 05: Configuring the Workstation

- Install sample software
- Configure workstation – minicom and networking
- Connect and power up target

Module 06: Building and Running Target Applications

- Flash memory and file systems
- Accessing peripherals from User space
- Data acquisition example

Module 07: Debugging Applications: On the Target and on the Workstation

- Debugging on the target
- Building a high-level simulation
- Debugging on the workstation

Module 08: Multi-threading

- The fork() function
- Posix threads
- Thermostat with threads

Module 09: Networking Part 1

- The “sockets” model
- Simple client/server example
- Networked thermostat

Module 10: Networking Part 2

- Multiple client connections
- Embedded web server

Module 11: Graphics Device Driver

- What's a device driver?
- ANSI escape sequences
- The console driver
- The framebuffer driver
- Display example

Module 12: Configuring and Building the Linux Kernel Part 1

- The kernel source tree
- Patching software
- Configuring the kernel - xconfig
- The .config file
- Building the kernel
- Booting the new kernel over TFTP

Module 13: Configuring and Building the Linux Kernel Part 2

- Kconfig language
- Patching the source
- Cross-compilation
- Booting the new kernel over the network

Module 14: Busybox and U-boot

- Configuring Busybox
- Configuring and Building Busybox
- Building Busybox
- Configuring U-boot

Module 15: Booting the Application, Getting Ready to Ship

- How Linux boots up
- The init process and inittab file
- Loading the kernel to NAND flash
- Loading the root file system to NAND flash

Module 16: Wrapping Up

- Course Summary
- Website references
- Book recommendations