



Linux

Installing, programming and writing drivers

Industrial applications are more and more often performed using an embedded version of Linux. In addition, the very specific environment in which run these systems sometimes make it necessary to adapt the Linux installation to the hardware environment.

Ac6-training trainings not only teach you how to build applications on embedded Linux, but also how to adapt the operating system to your hardware or environment when the need arises.

All our Linux trainings are using the integrated development environment [System for Workbench Linux - Basic Edition](#), which is provided for free to all our students to enable them to continue, after the training, to work in a friendly, efficient and mastered environment.

We also offer ([see the "SysAdmin" tab](#)) a range of training to help you set up and administer a Linux infrastructure of workstations and servers.

You can see detailed course descriptions of the various trainings by using the above navigation bar. You can also click on course identifiers in the following course briefs hereafter.

Main Courses

D0 - Linux user mode programming Programming Embedded Linux Applications for Linux

D1 - Embedded Linux with Buildroot and Yocto Building and installing an embedded and real-time Linux platform
Installing Linux on an embedded system is a common yet often difficult task. Many Open Source tools are supposed to make things easier, but must be properly controlled to obtain satisfactory results. This training presents you the most common of these tools (crosstool-NG, buildroot, OpenEmbedded, Yocto, System Workbench for Linux ...) and how to use them. The problems due to porting the u-boot bootloader and implementing a Linux BSP are also discussed as well as a porting methodology.

D1S - Embedded Linux with Ac6 System Workbench Implementing Linux on Embedded Systems

Installing Linux on an embedded system is a common yet often difficult task. Ac6 System Workbench was designed to make things easier and to be easily extended. This training presents you the architecture and needs of an Embedded Linux platform and explains how to build it using System Workbench for Linux. The problems due to porting the u-boot bootloader and implementing a Linux BSP are also discussed as well as a porting methodology.

D1Y - Embedded Linux with Yocto Building embedded Linux platforms using Yocto
Installing Linux on an embedded system is a common yet often difficult task. The Yocto project is meant to

D3 - Linux Drivers Writing Linux Drivers
This course covers the various techniques needed to write Linux (2.6 and 3.x) drivers, bus management (PCI, I2C, SPI, USB, CAN, etc.) and the tools to measure real-time performance.

D4 - Real-time Linux Real-time Linux with RT-Preempt patch and Xenomai
This course presents the various solutions for a real-time Linux and the tools to measure real-time performance.

D5 - Embedded GUI Graphical User Interfaces for Embedded Linux

D7 - Linux USB Drivers, HotPlug and Power Management Writing drivers with hot-plug and power management
This course delves into the concepts of generic devices/drivers for Linux, the management of hotplug, as well as the management of power.

D8 - USB Linux Drivers Writing USB 2.0 and USB 3.0 host and gadget drivers on Linux
This course details the Linux driver model, the USB hotplug and power management architecture to write USB drivers.

Y1 - Yocto Project Development Building a Linux Embedded image using Yocto
Installing Linux on an embedded system is a common yet often difficult task. The Yocto project is meant to

Y12 - Comprehensive Yocto Project Usage This course is the combination of the [Yocto Project](#) and [Project](#)

Expert course; it is intended for engineers that need to fully understand the Yocto build environment and be

Y2 - Yocto Project Expert Advanced Yocto Project usage and adaptation

This course expects you to already know how to build a Linux platform using Yocto (see our

Additional Courses

E1 - Eclipse (TM) Utilisation de l'environnement de développement Eclipse pour C, C++ et Java

G1 - Android Installation Android installation on a hardware platform

Installing Android on a new platform is a complex process, you need to port first the Linux kernel then install

G2 - Android Programming Programming applications for the Android platform

Android was designed to allow quickly creating powerful and ergonomic interfaces for embedded, resource

G3 - Android Internals Android Frameworks and HAL Implementation

Installing Android on a new platform is a complex process requiring a deep understanding of the internals of

G5 - Android for Industrial System Control Building friendly interfaces for industrial systems with Andro

New industrial systems need sophisticated and ergonomic user interfaces. Building these with traditional

BT1 - Real Time and Multi-Core programming Programming Linux real-time and multi-core systems, av

Real-time and embedded code, especially targetting multicore processors, cannot be effectively tested, it m

SW1 - System Workbench for Linux Building embedded Linux systems using System Workbench

Installing Linux on an embedded system is a common yet often difficult task. The System Workbench was c