

Android Porting

Android is not just for smart phones. It is an open source operating system that can be embedded into a wide range of target hardware, with applications such as point of sale, test and measurement, industrial control, and information kiosks. This four day course will teach you how to create a custom Android platform from scratch, based on code from the Android Open Source Project (AOSP). The course is presented in modules, each with a practical session where you get the chance to try out the techniques described earlier. You will build up a functional embedded Android system, using a BeagleBone Black development board as the example target device. You will learn how to adapt the Android hardware abstraction layer to work with your target platform, how to add your own code and packages to the final system image and how to encapsulate the end-user experience into a single function device working in what is commonly termed “kiosk mode”.

Duration

4 days

Audience

This course is intended for developers who are starting out on a new Android project or who need to understand and modify an existing one

Prerequisites

- Good knowledge of C/C++
- Linux command-line and development tools such as make
- Familiarity with Java

Course materials

All students will receive:

- Printed and electronic copies of the presentations and lab notes
- Worked solutions to the problems

About the trainer



Chris Simmonds has a wealth of experience in customising Android for embedded systems. He has been running training courses and workshops in Android since 2010 and has trained teams at many well-known organisations in the UK, Europe, USA, South America and SE Asia. He is the author of the book “Mastering Embedded Linux Programming”, and is a frequent presenter at open source and embedded conferences, including Embedded Linux Conference and Embedded World. You can see some of his work on the "Inner Penguin" blog at www.2net.co.uk

Enquiries and bookings

Please email training@2net.co.uk or call +44 (0)7788 130719

Android Porting

Course outline

The Android Open Source Project

- The steps to port Android to a new platform
- Getting the AOSP code
- Building Android from source

Creating a new Android product

- Setting basic hardware characteristics
- Selecting which packages to build into the final images
- Adding a new product to the “lunch” menu

The Android kernel

- Android-specific kernel features
- Obtaining vendor kernel code
- Building a custom kernel into target images

Bootloaders and bootstrap

- Booting Android: boot image blobs
- Flashing images using fastboot
- The init program and init.*.rc scripts
- Starting native services: how to add your own

Tracing and debugging

- Profiling code execution using perf
- Profiling using Android Systrace
- Using gdbserver to debug native code

The Android build system

- Understanding the Android.mk make file
- Creating your own packages

The Android framework

- Android architecture
- Binder

- Android services
- Interface between framework and native layer: JNI

The Hardware Abstraction Layer

- The role of the libhardware libraries
- A step-by-step walk through of a HAL library

The graphics stack

- Surfaces and SurfaceFlinger
- The hwcomposer HAL library
- The gralloc HAL library
- The vendor OpenGL ES libraries

Human input devices

- The input layer: touch screens, mice, keyboards and keypads
- Adding input devices

Sound and vision

- The audio HAL library
- Configuring ALSA sound drivers
- Video sources: camera devices and other inputs
- Using OpenMAX plugins to add custom codecs

Networking

- Configuring network interfaces
- Mobile data, WiFi and Ethernet

Power management

- Interaction between Power Manager and the kernel
- Power states
- Writing power-aware kernel drivers