G5 - Android for Industrial System Control

Building friendly interfaces for industrial systems with Android

Objectives

- Discover the Android system architecture.
- Learn to configure and compile the Android sources to get a working system.
- Understand the Android SDK and NDK
  - Learn how to build a simple application
  - Learn the basics of man-machine Interface with Android
  - Discover how to interface Java code and native code
- Explore the Android source code architecture
  - The Android init process
  - System services
  - The Android Hardware Abstraction Layer

Labs are conducted on i.MX6 or i.MX8 boards
We use the last open source version of Android, as available on the board.
For on-site trainings, if suitable Linux workstations are not available, we provide virtual machine images for VirtualBox; the only requisite is then a recent 64bit PC with at least 8Gb of RAM and 100Gb of free disk space.

Who should attend this course?

- Engineers that must develop Android applications to control industrial systems.
- Architects that want to understand the benefit they may obtain from using Android in their products.

What you will be able to do after the training

- Install Android on an embedded platform
- Interface an Android platform with an external system
- Create a typical Android Embedded application.

Prerequisite

- Basic Linux user experience
- Basic C (or C++) programming skills
- Due to the high degree of advanced Java techniques being used, good Java development skills are mandatory
  - See our cours L4G - Java pour Androidfor quickly learning the necessary Java techniques

Course environment

- Printed course material (in English).
- One Linux PC for two trainees.
- One target Android platform (dual Cortex/A9) for two trainees
- External sensor hardware on an Arduino board connected in USB with the Android platform

Environnement du cours

- Cours théorique
  - Support de cours imprimé et au format PDF (en anglais).
  - Le formateur répond aux questions des stagiaires en direct pendant la formation et fournit une assistance technique et pédagogique.
- Activités pratiques
  - Les activités pratiques représentent de 40% à 50% de la durée du cours.

    - Elles permettent de valider ou compléter les connaissances acquises pendant le cours théorique.
    - Exemples de code, exercices et solutions
    - Un PC (Linux ou Windows) par binôme de stagiaires (si plus de 6 stagiaires) pour les activités pratiques avec, si approprié, une carte cible embarquée.
    - Le formateur accède aux PC des stagiaires pour l'assistance technique et pédagogique.
- Une machine virtuelle préconfigurée téléchargeable pour refaire les activités pratiques après le cours
- Au début de chaque demi-journée une période est réservée à une interaction avec les stagiaires pour s'assurer que le cours répond à leurs attentes et l'adapter si nécessaire

Audience visée

- Tout ingénieur ou technicien en systèmes embarqués possédant les prérequis ci-dessus.

Plan du cours

First Day

Android Architecture Overview

- Linux and Android
- Android licensing

The Android Build System

- The Android code base
- Building Android
    - The Android build environment
    - The Android build system
    - The Android.mk files
- Adding new components to the build system
    - Java components
    - Native components
    - Applications

Exercise : Compiling the Android platform

Android Application Structure

- Structure of an Android Application
- Android application components
    - Activity
    - Service
    - Broadcast receiver
    - Content provider
- Manifest file
    - Application components declaration
    - Permissions

Exercise : Hello world application

- User interface configuration
    - Depending on the language
    - Depending on screen characteristics (dimensions, orientation&)

Exercise : Multilingual Hello world (Deutsch-English-Français)

The Android System Initialization

- Android properties

- The Android initialization
  - Structure of the init process
  - The Android initialization language
- The Dalvik Java virtual machine
  - The Dalvik machine structure
  - The Dalvik bytecodes
  - The Dalvik zygote process

Exercise : Tailoring Android initialization to start additional system daemons

Second Day

Activities and user interface

- Activities life cycle
- Activity callbacks
  - onCreate
  - onStart&
- Intents and Intents filter
  - The Intent class Intent
  - Declaring Intent filters in manifest files
- Activity invocation with and without results
  - startActivity
  - startActivityForResult
- Tasks (activities stack) and navigation between activities

Exercise : Writing a simplified parameter entry application

Defining user interface layout

- Layouts
  - Layout kinds
  - Components properties related to layouts
- Resources
  - Strings
  - images
  - layouts&
- Views
  - Buttons, labels and edition fields
  - View instantiation from a resource
- Specialized views
  - ListView
  - Data binding (Adapter class and subclasses)
- User Input
  - Touch screen and keyboard
  - Software keyboard management
- Dialogs and User notifications
  - Dialog box
  - Status Bar
  - Toast

Exercise : Writing a simple Command and Control application

The Android Sensors

- Sensors in Android
  - The sensor types
  - The Sensor Manager
  - Accessing Sensors
- Framework Architecture

    - Sensor discovery
    - Sensor Calibration

Exercise : Getting and displaying a sensor value (temperature...)

Third day

Android as a Distributed System

- The Android Binder architecture
- Binder implementation
    - The AIDL language
    - The AIDL tool
    - Binder Java classes
- Writing Application Services
- System services
    - What is a system service
    - Static and context-dependent services
    - Structure of a system service
    - Adding a new system service
    - The system ServiceManager process

Exercise : Coding a service to control an external device

Android Native Interface

- The Android NDK
    - Defining Java methods in C++
    - JNI for Android
- Integrating native code in a package
    - Using the NDK from Eclipse
    - Debugging native code

Exercise : Displaying data fetched from an external device

Advanced User Interface

- User interface and multithreading
    - Accessing views from another thread

Exercise : Multi-threaded user interface with buttons and progress bars

- Custom control creation
    - By deriving directly the View class
    - By deriving an existing view
- 2D Drawing
    - Canvas and Shapes
    - Drawing from the main thread
    - Drawing from another thread

Exercise : Displaying a graph of sensor values

Data management

- Storage
    - Shared preferences
    - Internal storage
    - External storage
    - SQLite
- Content provider
    - Communication with a content provider
    - Implementing a content provider

Exercise : Logging data fetched from the external device and displaying historical data

Fourth Day

Broadcast Receivers

- Installing a Broadcast Receiver
  - Static creation of broadcast receivers
  - Dynamic instantiation and registration
- Broadcasting intents
  - Normal broadcast
  - Ordered broadcast
- Using PendingIntent in broadcast receivers
- System broadcasted events

Exercise : Handling process alarms in a custom broadcast receiver

Networking

- Connections management
- Sockets
- HTTP requests
- WebView control
- Web Services

Exercise : Socket communications with a distant management application

The Hardware Abstraction Layer

- Why a HAL?
- HAL Component Structure
  - Defining HAL components
  - Loading and using HAL component
- The standard HAL components
  - Graphics
  - Audio
  - Camera
  - Bluetooth
  - GPS
  - Sensors

Exercise : Create a simple HAL component