



## D1L - Linux embarqué NXP avec LTIB

### Linux pour les systèmes embarqués Freescale à base ARM et PowerPC

#### Objectifs

- Concevoir son propre système Linux avec LTIB
- Utiliser les outils de développement natifs ou croisés
- Personnaliser, réduire et configurer le noyau Linux
- Embarquer Linux sur Flash EPROM
- Détailler la procédure de boot Linux
- Monter un système de fichier distant
- Booter un noyau Linux distant

#### Matériel

- Un PC Linux par binôme
- Une carte embarquée par binôme :
  - 1 carte NXP i.MX31
  - 1 carte simulée (Qemu) à base d'ARM 926 (carte de référence "Versatile" de chez ARM)
- Support de cours
- CDRom avec documentation et outils utilisés

#### Pré-requis

- Connaissance de Linux utilisateur

#### Course Environment

- Theoretical course
  - PDF course material (in English) supplemented by a printed version for face-to-face courses.
  - Online courses are dispensed using the Teams video-conferencing system.
  - The trainer answers trainees' questions during the training and provide technical and pedagogical assistance.
- Practical activities
  - Practical activities represent from 40% to 50% of course duration.
  - Code examples, exercises and solutions
  - For remote trainings:
    - ▶ One Online Linux PC per trainee for the practical activities.
    - ▶ The trainer has access to trainees' Online PCs for technical and pedagogical assistance.
    - ▶ QEMU Emulated board or physical board connected to the online PC (depending on the course).
    - ▶ Some Labs may be completed between sessions and are checked by the trainer on the next session.
  - For face-to-face trainings:
    - ▶ One PC (Linux ou Windows) for the practical activities with, if appropriate, a target board.
    - ▶ One PC for two trainees when there are more than 6 trainees.
  - For onsite trainings:
    - ▶ An installation and test manual is provided to allow preinstallation of the needed software.
    - ▶ The trainer come with target boards if needed during the practical activities (and bring them back at the end of the course).
- Downloadable preconfigured virtual machine for post-course practical activities
- At the start of each session the trainer will interact with the trainees to ensure the course fits their expectations and correct if needed

## Target Audience

- Any embedded systems engineer or technician with the above prerequisites.

## Evaluation modalities

- The prerequisites indicated above are assessed before the training by the technical supervision of the trainee in his company, or by the trainee himself in the exceptional case of an individual trainee.
- Trainee progress is assessed in two different ways, depending on the course:
  - For courses lending themselves to practical exercises, the results of the exercises are checked by the trainer while, if necessary, helping trainees to carry them out by providing additional details.
  - Quizzes are offered at the end of sections that do not include practical exercises to verify that the trainees have assimilated the points presented
- At the end of the training, each trainee receives a certificate attesting that they have successfully completed the course.
  - In the event of a problem, discovered during the course, due to a lack of prerequisites by the trainee a different or additional training is offered to them, generally to reinforce their prerequisites, in agreement with their company manager if applicable.

## Plan

### 1er jour

## Architecture de Linux

- Présentation de Linux
- Licences GPL et open source
- Distributions Linux
- Architecture et modularité de Linux

## Les outils Linux pour l'embarqué

- Les firmware/bootloader (Uboot, Redboot,...)
- Les bibliothèques adaptées à l'embarqué (uClibc, dietlibc,...)
- Les IHM adaptées à l'embarqué (miniGUI, Qtopia, Nano-X,...)
- Busybox, le "couteau suisse" de Linux embarqué
- Les distributions spécialisées
  - Commerciales (MontaVista, Sysgo, Timesys, Windriver,...)
  - Open Source (ELDK, Koan, RTAI,...)

## Les chaînes de compilation croisée

- Chaînes complètes (ELDK, CodeSourcery, ...)
- Outils de compilation de chaînes (Crosstool-ng, Buildroot, ...)
- Compilation manuelle

*Exercise: Construction d'une chaîne avec Crosstool-ng, avec Buildroot. Etude des différences.*

### 2ème jour

## Création du noyau Linux pour l'embarqué

- Téléchargement des sources
- Etude du Makefile du noyau
- Patch du noyau
- Configuration du noyau
- Compilation native et croisée du noyau Linux et des modules
- Installation des modules et du noyau

## Etude du support des MTD dans Linux (Memory Technology Devices)

- Les mémoires de type NOR
- Les mémoires de type NAND

*Exercice: Patch d'un noyau "vanilla" afin de l'adapter à la carte i.MX31*

*Exercice: Configuration et compilation du noyau pour la carte i.MX31*

## Création/modification du Board Support Package (BSP)

- Etude des BSP ARM
  - Les numéros de machine ARM
  - Organisation des fichiers sources
  - Les Makefiles
  - Les fichiers de configuration
  - La struct machine\_desc

*Exercice: Modification du BSP pour le support et le partitionnement d'une mémoire NAND*

## Création du système de fichier racine

- Périphériques, programmes, bibliothèques...
- Installation des modules
- Recherche et installation des bibliothèques dynamiques nécessaires
- Vérification de la cohérence du système de fichier
- Utilisation de l'outil LTIB pour générer un système de fichiers racine embarqué pour cartes NXP.
  - Paramétrage et configuration
  - Sélection des packages constituant le système.
  - Modification de package
  - Ajout de package

*Exercice: Création d'un système fonctionnel avec LTIB*

*Exercice: Ajout de packages dans LTIB*

## 3ème jour

## Choix du type de système de fichier

- Ramdisk/initrd
- Systèmes de fichier adapté aux mémoires flashes (JFFS2, UBIFS, ...)
- Systèmes de fichier adaptés à l'embarqué (ROMFS, CRAMFS, ...)
- Systèmes de fichiers compatibles "desktop" (EXT2, FAT)

## Boot de Linux

- Partitionnement d'un disque Linux
- Bootloaders (lilo, grub, syslinux, RedBoot...)
- Paramètres du noyau au boot
- Etapes de chargement et de démarrage du noyau
- Choix des programmes lancés au démarrage (runlevels, services, suppression de l'IHM...)

*Exercice: Partitionnement dynamique de la NAND*

*Exercice: Flashage du noyau et du système de fichier*

*Exercice: Boot depuis la mémoire flash*

## 4ème jour

## Développement croisé

- Préparation de la machine hôte
  - Services réseau (DHCP, TFTP)

- Partage du système de fichier (NFS)
- Utilisation de la chaîne de développement croisé LTIB
- Etude des outils de développement (gss, gdb, Les Makefiles, ...)
- Utilisation de l'IDE Eclipse pour la programmation et le debug croisé
- Préparation de la cible
  - Téléchargement d'un noyau par TFTP
  - Démarrage sur un système de fichier distant (diskless)
  - Utilisation de gdbserver pour le debug distant
- Utilisation de Qemu pour simuler une carte complète

*Exercise: Démarrage de la carte IMX31 avec un système Linux LTIB distant.*

*Exercise: Démarrage du système Linux LTIB à partir d'un émulateur ARM (Qemu).*

*Exercise: Compilation, test et debug distant d'une application embarquée.*

*Exercise: Compilation et installation de modules noyau indépendants*

*Exercise: Compilation croisé de packaging*

## Renseignements pratiques

**Duration : 5 days**

**Cost : 2200 € HT**