

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

Environnement du cours

- Cours théorique
 - Support de cours au format PDF (en anglais) et une version imprimée lors des sessions en présentiel
 - Cours dispensé via le système de visioconférence Teams (si à distance)
 - 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
 - Pour les formations à distance:
 - ▶ Un PC Linux en ligne par stagiaire pour les activités pratiques, avec tous les logiciels nécessaires préinstallés.
 - ▶ Le formateur a accès aux PC en ligne des stagiaires pour l'assistance technique et pédagogique
 - ▶ Certains travaux pratiques peuvent être réalisés entre les sessions et sont vérifiés par le formateur lors de la session suivante.
 - Pour les formations en présentiel:
 - ▶ Un PC (Linux ou Windows) pour les activités pratiques avec, si approprié, une carte cible embarquée.
 - ▶ Un PC par binôme de stagiaires s'il y a plus de 6 stagiaires.
 - Pour les formations sur site:
 - ▶ Un manuel d'installation est fourni pour permettre de préinstaller les logiciels nécessaires.
 - ▶ Le formateur vient avec les cartes cible nécessaires (et les remporte à la fin de la formation).
- Une machine virtuelle préconfigurée téléchargeable pour refaire les activités pratiques après le cours

- Au début de chaque session (demi-journée en présentiel) 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.

Modalités d'évaluation

- Les prérequis indiqués ci-dessus sont évalués avant la formation par l'encadrement technique du stagiaire dans son entreprise, ou par le stagiaire lui-même dans le cas exceptionnel d'un stagiaire individuel.
- Les progrès des stagiaires sont évalués de deux façons différentes, suivant le cours:
 - Pour les cours se prêtant à des exercices pratiques, les résultats des exercices sont vérifiés par le formateur, qui aide si nécessaire les stagiaires à les réaliser en apportant des précisions supplémentaires.
 - Des quizz sont proposés en fin des sections ne comportant pas d'exercices pratiques pour vérifier que les stagiaires ont assimilé les points présentés
- En fin de formation, chaque stagiaire reçoit une attestation et un certificat attestant qu'il a suivi le cours avec succès.
 - En cas de problème dû à un manque de prérequis de la part du stagiaire, constaté lors de la formation, une formation différente ou complémentaire lui est proposée, en général pour conforter ses prérequis, en accord avec son responsable en entreprise le cas échéant.

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

Exercice : 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 flashs (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

Exercice : Démarrage de la carte I.MX31 avec un système Linux LTIB distant.

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

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

Exercice : Compilation et installation de modules noyau indépendants

Exercice : Compilation croisé de paquetage

Renseignements pratiques

Renseignements : 5 jours