

LINUX EMBARQUÉ

4 jours

Réf. : 5125

OBJECTIFS

Ce stage permet aux développeurs en informatique, confrontés aux problèmes de portage d'un noyau Linux et des applications sous cible linux embarqué, de pouvoir concevoir une distribution sur mesure, optimisée à l'aide des différentes solutions de configuration Linux et de mettre en œuvre une chaîne de développement croisé complète.

Cette formation Linux embarqué aborde les concepts du portage d'un OS Linux sur cible embarquée, par un tour d'horizon des différentes distributions et les approfondit par l'étude :

- des caractéristiques et architectures des systèmes Linux embarqué,
- de la mise en œuvre d'une chaîne de développement croisé,
- de la compilation d'un noyau et installation sur ROM/FLASH,
- des packages et configuration de l'installation sur différents types d'architecture matérielle x86 et ARM9.

PRE-REQUIS

Cette formation est particulièrement adaptée aux techniciens et ingénieurs, confrontés aux problèmes de portage de solutions linux sur systèmes embarqués.

PROGRAMME

PREMIÈRE JOURNÉE

Distributions Linux Embarqué

- Définition des systèmes embarqués
- Projets existants: MontaVista, Bluecat linux, uCLinux, eldk, PeeWee Linux...
- Méthodes, outils et chaîne de développement croisé

Noyau Linux - Présentation

- Vue d'ensemble du système et rôle du noyau
- Historique, numérotation des versions
- Architectures matérielles supportées - Support processeurs et File system
- Spécificités des noyaux 2.4 et 2.6

Noyau Linux – Configuration et Compilation pour l'Embarqué

- Structure des sources et modules du noyau
- Fichiers utilisés par les outils de configuration
- Configuration, optimisation et Compilation d'un noyau
- Compilation croisée - Méthodes, outils et chaîne de développement croisés,
- Choix de bibliothèques : LibC : glibc, uClibc, NewLibC

Travaux pratiques

- Création d'une chaîne de développement croisé pour cible ARM 9
- Configuration et compilation d'un noyau 2.6 « patché » pour cible ARM 9

DEUXIÈME JOURNÉE

Portage d'une image système Linux embarqué

- Systèmes de fichier et ramdisk initrd
- Paramétrage du chargeur de démarrage « bootloader »,
- CompactFlash, DiskOnChip, Flash Disk, ...
- Mise au point par port série, par réseau, par sonde JTAG (Abatron BDM BDI2000)
- Portage d'un Shell et d'outils d'administration (BusyBox ...)

Méthodes et outils de validation

- Portage d'une chaîne de débogueur à distance : serveur gdb
- Installation d'un système bootable via réseau et montage nfs

Travaux pratiques

- Mise en œuvre d'une configuration de bootloader sur plateforme ARM9 avec U-BOOT
- Réalisation d'une image Linux embarqué pour cible x86 PC104 bootable sur mémoire Compact Flash
- Mise en œuvre d'une chaîne de développement (débogueur embarqué) croisé pour cible ARM 9

TROISIÈME JOURNÉE

Services et configurations

- Script de démarrage et de configuration du système Linux embarqué
- Chargement dynamique de pilotes de périphériques
- Installation des services réseaux : Inetd, Rsh, telnet, Nfs
- Outils de configuration Buildroot

TROISIÈME JOURNÉE (suite)

Environnement utilisateur sous Linux embarqué

- Console série
- Gestion de sessions utilisateurs
- Connection réseau : remote shell rsh, telnet, ...
- Portage et configuration d'environnement graphique X: nanoX, XFree, QtE/QTopia, MicroWindows, ...

Travaux pratiques

- Réalisation Installation et configuration d'un système linux embarqué x86 PC104 avec prise en charge réseau

QUATRIÈME JOURNÉE

Introduction au développement de pilotes

- Pilotes de périphériques sous Linux
- Introduction à la programmation en mode noyau
- Architecture d'un module simple
- Programmation de pilote de périphérique

Introduction aux extensions Temps réel embarquées

- Temps reel souple: interface POSIX, extension API PThread , NPTL,
- Temps reel dure: Patches low-latency, préemptifs, Temps Réel RTAI/RTLinux
- Introduction à la programmation temps reel RTAI

Travaux pratiques

- Construction et compilation d'un pilote de carte E/S PC104 Linux embarqué
- Programmation d'une tâche temps réel de traitement périodique à 10Khz

© CenoSYS 2010-01