

L'utilisation de Linux dans les environnements embarqués sur des systèmes restreints nécessite une bonne connaissance des mécanismes sous-jacents, de la configuration du noyau, de l'organisation des bibliothèques et de la mise en œuvre des services et applications utilisateurs.

Ce cours vous propose de construire un système Linux complet en utilisant une carte à processeur ARM.

Il existe aujourd'hui deux approches pour construire un système embarqué : l'utilisation de Buildroot ou celle de Yocto. Cette formation est orientée autour de Buildroot, plus simple à maîtriser que Yocto.

De la configuration bas-niveau du système (drivers, ordonnanceur, système de fichiers...) aux applications utilisateur (serveurs, IHM, graphique...) en passant par les outils de mise au point et de débogage, vous découvrirez l'imbrication et le rôle de chaque composant de votre système Linux embarqué.

Les travaux pratiques pour ce cours se déroulent sur des cartes « Raspberry Pi » (une carte par participant) mises à disposition par Logilin.

Organisation

Durée : 3 jours (21 heures).

Pré-requis : Connaissance de Linux (utilisateur) et notions de langage C.

Conseil cursus : Les formations « Temps-réel sous Linux et Xenomai » et « Écriture de Drivers pour Linux » représente un bon complément à ce cours.

Thèmes abordés

- **Création d'un système Linux embarqué** : outils de développement, compilation croisée, utilisation de Buildroot.
- **Personnalisation d'un système embarqué** : ajustement du système, des services, du réseau, paramétrage du noyau.
- **Développement du code métier** : chaîne de compilation, débogage à distance, optimisation.

Plan détaillé

I - Créer un système Linux embarqué

Linux embarqué

Spécificités du développement embarqué, architectures, *cross-compilation*, *build systems*, utilisation de Buildroot.

Composition d'un système Linux embarqué

Aspects matériels, composants logiciels, noyau, *drivers*, espace utilisateur.

Boot du système

Firmware, *bootloader*, *kernel*, processus *init*, partitionnement, démarrage des services du système.

Travaux pratiques

Utilisation de Buildroot pour créer une première image et test sur la plateforme cible.

II - Personnalisation d'un système embarqué

Configuration du système avec Buildroot

Utilisateurs et mots de passe, messages, arborescence, ajouts de scripts personnalisés, configuration de Busybox, ajout de commandes.

Paramétrage du réseau

Configuration statique ou utilisation de DHCP, Services SSH, HTTP, NTP.

Noyau Linux

Configuration du noyau Linux, choix des drivers, licences libres.

Travaux pratiques

Paramétrage de Buildroot pour mettre en évidence chacun des éléments de configuration abordés.

III - Développement du code métier

Outils de développement

Chaîne de compilation croisée, extraction et installation. Utilisation des *makefiles*,

Débogage distant

Utilisation de GDB et GDBserver, débogage à distance, utilisation des fichiers *core*.

Optimisation et mise au point

Outils *gcov*, *gprof*, *strace*, *ltrace*, *valgrind*...

Travaux pratiques

Utilisation de la chaîne de cross-compilation extraite de Buildroot. Débogage d'une application embarquée sur la plateforme cible. Utilisation des outils libres présentés.

Conclusion

Discussions libres sur l'ensemble des thèmes abordés.

Travaux pratiques

Expérimentations libres suivant les demandes des participants.