



BSP UBoot et Linux embarqué : construire son propre système

Lien :

<https://innov-systems.com/formation/bsp-uboot-et-linux-embarque-construire-son-propre-systeme>

 DURÉE
5 jours (35h)

 RÉFÉRENCE
RIS26

 CATÉGORIE
Linux Embarqué

OBJECTIFS DE LA FORMATION

À l'issue de cette formation, vous serez capable de :

- ✓ Comprendre l'architecture d'un système Linux
- ✓ Créer et utiliser une chaîne de compilation croisée
- ✓ Apprendre à installer Linux sur votre cible matérielle et écrire un BSP
- ✓ Etre capable de développer efficacement du code applicatif métier sur un système Linux embarqué
- ✓ Connaître les mécanismes d'ordonnancement temps partagé et réel souple

POUR QUI ?

- ✓ Architecte ou ingénieur



☰ Programme détaillé

1 / Architecture de Linux

- Linux : Histoire. Gestion de version
- Les diverses licences utilisées par Linux (GPL, LGPL, etc)
- Distributions Linux
- Architecture et modularité de Linux

2 / Les chaînes de compilation croisée

- Chaînes de compilation croisée pré-compilées
- Outils de génération de chaînes de compilation croisée : Crosstool-ng, Buildroot
- Compilation manuelle de chaîne de compilation croisée

3 / Outils libres de développement embarqué

- Outils GNU : Compilation, débogage et mise au point
- Platesformes hôte et cible, processeurs et émulateur

4 / Introduction à Yocto

- Présentation de Yocto
- Histoire
- Yocto, Open Embedded et Poky
- Objectif du projet Yocto
- Les principaux projets
- Architecture Yocto
- Aperçu
- Recettes et classes
- Les tâches

5 / Créer le noyau Linux embarqué

- Télécharger un code source stable
- Configurer le noyau
- Compiler le noyau et ses modules
- Installer le noyau et les modules
- Application de patches
- Transférer l'image du noyau
- Paramétrer le démarrage du kernel
- Types de système de fichiers
- Formater et préparer une arborescence

6 / Le BSP Linux

- Architecture du BSP Linux
- Structure générale
- Le BSP ARM
- Le système de compilation de Linux
- Définir et initialiser la carte
- Programmation (platform, i2c, spi, ...)
- En utilisant le Flattened Device Tree

7 / Le boot loader U-Boot

- Les bootloaders (Uboot, Redboot, barebox...)
- Introduction à U-Boot
- Installer U-Boot
- Booter par TFTP et par NFS
- Accéder aux flashes
- Accéder aux systèmes de fichier (NFS, FAT, EXTx, JFFS2...)
- Organisation des partitions. Gestion de versions

8 / Espace utilisateur

- Busybox, les utilitaires basiques
- Configurer et compiler Busybox
- Configurer le réseau et des serveurs
- Authentification des utilisateurs et connexion distante

9 / Code métier

- Outils de développements : compilateurs, interpréteurs, IDE
- Débogueur, profileur, couverture de code
- Bibliothèques statiques et dynamiques
- Gestion de mémoire sous Linux

10 / Drivers et modules spécifiques

- Le principe des drivers
- Vue d'ensemble des principes de la programmation noyau
- Vue d'ensemble de la programmation de drivers en mode caractère

11 / Ordonnancement sous linux

- Multi-tâche sous Linux
- Systèmes multi-processeurs symétriques (SMP)

12 / Temps réel souple

- Temps réel : principes
- Fifo et Round-robin
- Timers
- Interruptions
- Problèmes temps-réel classiques :
- Lancement de tâches en parallèle

- Inversion de priorité
- Reprise de mutex
- Création de processus et de threads temps-réel
- Mesure de précision des timers
- Effet de la préemptibilité du noyau sur la latence des interruptions
- Examen d'inversion de priorité. Héritage de priorité. Test de reprise de mute

13 / Temps réel amélioré avec linux-rt

- Patch Linux-rt :
- Principe
- Patch d'Ingo Molnar et Thomas Gleixner
- Compilation du noyau
- Prémption totale
- Threadeds interrupts : Configuration de la priorité des handlers d'interruptions
- Instrumentation et mesures : Outils RT-test et cyclicttest
- Compilation d'un noyau après application du patch Linux-rt. Vérification de la préemptibilité
- Utilisation de cyclicttest et comparaison avec le noyau standard
- Comparaison du comportement des exemples du chapitre précédent

14 / Temps réel strict pour linux - Extension Xenomai

- Concepts de temps-réel strict
- Présentation de l'extension Xenomai

Approche pédagogique

- ✓ Support Ecrit et Projection
- ✓ Exposés Interactifs, Podcasts et Vidéos
- ✓ Brainstorming et Jeux de Rôle
- ✓ Cas Pratiques et Labs inclus pour leur impact opérationnel
- ✓ Test de Validation des Acquis des Connaissances

📅 Prochaines dates programmées

📅 27 au 31 Juil. 2026

📍 Présentiel - Casablanca

📅 21 au 25 Sep. 2026

📍 Distanciel

📅 16 au 20 Nov. 2026

📍 Distanciel

📅 Autres dates possibles sur demande. Contactez-nous pour organiser une session intra-entreprise.

🔄 Réservation & Renseignements

📞 **Téléphone** : +212 522 247 210

✉️ **Email** : contact@innov-systems.com

🌐 **Web** : <https://www.innov-systems.com>

Document généré le 29/06/2026 — Réf : RIS26
Innov Systems — Tous droits réservés