



# Formation temps-réel sous Linux avec *PREEMPT\_RT*

Formation sur site, 2 jours  
Dernière mise à jour : 13 January 2025

<b>Titre</b>	<b>Formation temps-réel sous Linux avec <i>PREEMPT_RT</i></b>
<b>Objectifs opérationnels</b>	<ul style="list-style-type: none"><li>• Être capable de comprendre et de maîtriser les caractéristiques d'un système d'exploitation temps-réel</li><li>• Être capable de télécharger, compiler et utiliser le patch <i>PREEMPT_RT</i></li><li>• Être capable d'identifier et de benchmarker la plateforme matérielle en terme de caractéristiques temps-réel</li><li>• Être capable de configurer le noyau Linux pour un comportement déterministe</li><li>• Être capable de développer, de tracer et de déboguer des applications user-space temps-réel.</li></ul>
<b>Durée</b>	<b>Deux jours - 16 h (8 h par jour)</b>
<b>Méthodes pédagogiques</b>	<ul style="list-style-type: none"><li>• Présentations animées par le formateur : 50% de la durée de formation</li><li>• Travaux pratiques réalisés par les participants : 50% de la durée de formation</li><li>• Version électroniques de supports de présentation, des instructions et des données de travaux pratiques. Les supports sont librement disponibles sur <a href="https://bootlin.com/doc/training/preempt-rt">https://bootlin.com/doc/training/preempt-rt</a>.</li></ul>
<b>Formateur</b>	Maxime Chevallier <a href="https://bootlin.com/company/staff/maxime-chevallier/">https://bootlin.com/company/staff/maxime-chevallier/</a>
<b>Langue</b>	Présentations : Français Supports : Anglais
<b>Public visé</b>	Entreprises et ingénieurs intéressés dans le développement et le benchmarking d'applications et de drivers temps-réel pour un système Linux embarqué.



<b>Pré-requis</b>	<ul style="list-style-type: none"><li>• <b>Connaissance et pratique des commandes UNIX ou GNU/Linux</b> : les participants doivent être à l'aise avec l'utilisation de la ligne de commande Linux. Les participants manquant d'expérience sur ce sujet doivent se former par eux-mêmes, par exemple en utilisant nos supports de formation disponible à l'adresse <a href="https://bootlin.com/blog/command-line/">bootlin.com/blog/command-line/</a>.</li><li>• <b>Expérience minimale en développement Linux embarqué</b> : les participants doivent avoir une compréhension minimale de l'architecture d'un système Linux embarqué : rôle du noyau Linux par rapport à l'espace utilisateur, développement d'applications espace utilisateur en C. Suivre la formation <i>Linux embarqué</i> de Bootlin, disponible sur <a href="https://bootlin.com/training/embedded-linux/">bootlin.com/training/embedded-linux/</a>, permet de remplir ce pré-requis.</li><li>• <b>Niveau minimal requis en anglais</b> : <b>B1</b>, d'après le <i>Common European Framework of References for Languages</i>, pour nos sessions animées en anglais. Voir <a href="https://bootlin.com/pub/training/cefr-grid.pdf">bootlin.com/pub/training/cefr-grid.pdf</a> pour une auto-évaluation.</li></ul>
<b>Équipement nécessaire</b> <b>nécessaire</b>	<p><b>Pour les sessions en présentiel dans les locaux de nos clients, notre client doit fournir :</b></p> <ul style="list-style-type: none"><li>• Projecteur vidéo</li><li>• Un ordinateur sur chaque bureau (pour une ou deux personnes), avec au moins 8 Go de RAM et Ubuntu Linux 24.04 installé dans une <b>partition dédiée d'au moins 30 Go</b>.</li><li>• Les distributions autres que Ubuntu Linux 24.04 ne sont pas supportées, et l'utilisation de Linux dans une machine virtuelle n'est également pas supportée.</li><li>• <b>Connexion à Internet rapide et sans filtrage</b> : au moins 50 Mbit/s de bande passante en téléchargement, et pas de filtrage des sites Web et protocoles.</li><li>• <b>Les ordinateurs contenant des données importantes doivent être sauvegardés</b> avant d'être utilisés dans nos sessions.</li></ul>
<b>Modalités d'évaluation</b>	<p>Seuls les participants qui auront assisté à l'intégralité des journées de formation, et qui auront obtenu plus de 50% de réponses correctes à l'évaluation finale recevront une attestation individuelle de formation de la part de Bootlin.</p>



## Handicap

Les participants en situation de handicap qui ont des besoins spécifiques sont invités à nous contacter à l'adresse [training@bootlin.com](mailto:training@bootlin.com) afin de discuter des adaptations nécessaires à la formation.

## Plateforme matérielle pour les travaux pratiques

Une de ces cartes de STMicroelectronics :  
**STM32MP157A-DK1**, **STM32MP157D-DK1**,  
**STM32MP157C-DK2** ou **STM32MP157F-DK2**

- Processeur STM32MP157, double Cortex-A7, de STMicroelectronics
- Alimentée par USB
- 512 Mo DDR3L RAM
- Port Gigabit Ethernet port
- 4 ports hôte USB 2.0
- 1 port USB-C OTG
- 1 connecteur Micro SD
- Debugger ST-LINK/V2-1 sur la carte
- Connecteurs compatibles Arduino Uno v3
- Codec audio
- Divers : boutons, LEDs
- Écran LCD tactile (uniquement sur cartes DK2)



## 1 demi-journée

### Cours - Introduction au comportement temps-réel et au déterminisme

- Définition d'un système d'exploitation temps-réel
- Spécificités des systèmes multi-tâches
- Principaux patterns de verrouillage et de gestion des priorités
- Aperçu des systèmes temps-réel existants
- Approches pour apporter un comportement temps-réel à Linux



## Cours - Le patch *PREEMPT\_RT*

- Histoire et avenir du patch *PREEMPT\_RT*
- Améliorations temps-réel provenant de *PREEMPT\_RT* dans le noyau Linux officiel
- Fonctionnement interne de *PREEMPT\_RT*
- Gestion des interruptions : interruptions threadées, softirqs
- Primitives de verouillage : mutexes et spinlocks, spinlocks avec sommeil
- Modèles de préemption

## TP - Compiler un noyau Linux avec *PREEMPT\_RT*

- Télécharger le noyau Linux et appliquer le patch *PREEMPT\_RT*
- Configurer le noyau Linux
- Démarrer le kernel sur une plateforme matérielle

## 2 demi-journée

---

### Cours - Configuration et limites du matériel pour le temps-réel

- Interruptions et firmware
- Interaction avec les fonctionnalités de gestion d'énergie : gestion dynamique de la fréquence du CPU et états de sommeil
- DMA

### Cours - Outils : Benchmarking, Stress et Analyse

- Benchmarking avec *cyclictest*
- Stress du système avec *stress-ng* et *hackbench*
- L'infrastructure de *tracing* du noyau Linux
- Analyse de la latence et de l'ordonnancement avec *ftrace*, *kernelshark* ou *LTTng*



## TP - Outils : Benchmarking, Stress et Analyse

- Utilisation des outils de benchmark et de stress
- Techniques classiques de benchmarking
- Benchmarking et configuration de la plateforme matérielle

## 3 demi-journée

---

### Cours - Infrastructures du noyau Linux et configuration

- Bonnes pratiques pour le développement de drivers noyau Linux pour des systèmes temps-réel
- Politiques d'ordonnancement et priorités : *SCHED\_FIFO*, *SCHED\_RR*, *SCHED\_DEADLINE*
- Affinité CPU et IRQ
- Gestion mémoire
- Isolation des CPUs avec *isolcpus*

### Cours - Patterns de développement d'applications temps-réel

- API POSIX pour les applications temps-réel
- Gestion et configuration des threads
- Gestion mémoire : allocation mémoire et verouillage mémoire, gestion de la pile
- Patterns de verrouillage : mutexes, héritage de priorité
- Communication inter-processus (IPC)
- Signalisation



## 4 demi-journée

---

### TP - Débugger une application de démonstration

- Créer une application de démonstration déterministe
- Utiliser l'infrastructure de *tracing* pour identifier la source de latence
- Apprendre à utiliser l'API POSIX pour gérer les threads, le verouillage, la mémoire
- Apprendre à utiliser l'affinité CPU et configurer la politique d'ordonnancement

### Questions / réponses

- Questions et réponses avec les participants à propos des sujets abordés.
- Présentations supplémentaires s'il reste du temps, en fonction des demandes de la majorité des participants.