



## Real-Time Operating Systems

Comme toute application informatique, les applications industrielles ou embarquées doivent être adaptées au système d'exploitation qui les supportent. De plus, l'environnement souvent très spécifique dans lesquelles s'exécutent ces systèmes font qu'il est parfois nécessaire de les adapter à l'environnement matériel.

ac6-formation vous propose des formations vous permettant de réaliser des applications utilisant un système d'exploitation embarqué, mais aussi d'adapter ce système d'exploitation si le besoin s'en fait sentir.

FreeRTOS + lwIP - 5 jours  
CORSY: l'écosystème IoT dédié à l'équipement IoT les plus utilisés de la périoétrie au cloud (MQTT, MO-DL, MO-C, MO-S) aux protocoles attachés particulièrement critiques axés sur les dispositifs de sécurité à chaque niveau de la pile (MQTT, TLS, DTLS) et systèmes de communication associés. Ce cours explique comment combiner le matériel, la connaissance préalable de systèmes de communication basés sur des microcontrôleurs. Il nécessite une connaissance préalable de réseaux et de systèmes d'exploitation embarqués. **Renseignements**

FreeRTOS est un système d'exploitation temps réel (RTOS) léger conçu pour gérer les tâches dans des applications embarquées en temps réel. Le cours couvre les aspects tels que les concepteurs et la synchronisation et le memory management. Destiné aux développeurs ayant une bonne compréhension de base des systèmes temps réel, ce cours offre une solide fondation des applications en temps réel pour des solutions embarquées modernes. **Renseignements**

FreeRTOS sur un microprocesseur ARM Cortex-M4F: une configuration via Device Tree et Kconfig, modules matériels, sous-interruptions, TIM5 pour le système à haute performance. **Renseignements**

FreeRTOS sur un microprocesseur ARM Cortex-M4F: une configuration via Device Tree et Kconfig, modules matériels, sous-interruptions, TIM5 pour le système à haute performance. **Renseignements**

FreeRTOS sur un microprocesseur ARM Cortex-M4F: une configuration via Device Tree et Kconfig, modules matériels, sous-interruptions, TIM5 pour le système à haute performance. **Renseignements**

FreeRTOS sur un microprocesseur ARM Cortex-M4F: une configuration via Device Tree et Kconfig, modules matériels, sous-interruptions, TIM5 pour le système à haute performance. **Renseignements**