

#### Objectifs

Cette formation est réalisée autour de BSP LinRT Helium Yocto kirkstone 4.0, NXP Linux kernel 5.10 PREEMPT-RT sur le Kit Polis i.MX 8M Mini

- Maîtriser l'installation de l'environnement Yocto kirkstone 4.0
- Maîtriser les commandes usuelles Bitbake
- Gérer la configuration de build et de la Machine i.MX 8
- Ajouter et Mettre en oeuvre ses propres layers Yocto
- Construire et customisez son image Linux avec Yocto
- Développer et mettre au point des applications avec le SDK Yocto i.MX 8
- Configurer et adapter le noyau linux à sa plate-forme i.MX 8
- Adapter les drivers du noyau Linux pour i.MX 8
- Développer et mettre au point des applications Qt 6.4 avec le SDK Yocto i.MX 8

#### Pré-requis

Notre formation Yocto BSP pour processeurs NXP iMX 8 est adaptée aux développeurs en informatique et techniciens dans le domaine de Linux embarqué confrontés aux problèmes de portage d'applications Temps Réel sur système i.MX 8 sous Yocto.

Une connaissance des commandes UNIX usuelles et des notions de programmation en langage C sont nécessaires.

##### Projet Yocto

- Présentation du projet Yocto
- OpenEmbedded Core et poky
- Notion de Layers et de Recettes
- Les utilitaires GIT et repo
- Rôle de l'outil de construction bitbake

##### Yocto FSL Community

- Les layers spécifiques FSL
- Les différentes distributions FSL et FSLC
- Les images applicatives proposées

##### Travaux pratiques

- Mise en oeuvre d'une configuration de BSP avec choix des layers, configuration de la distribution et de l'image fslc-framebuffer à générer pour une machine spécifique : SBC phyBOARD- Polis i.MX 8M Mini

##### FSL Community BSP

- Fichier local.conf et réglages globaux
- Les différentes cibles NXP i.MX supportées
- Notion de MACHINE et de PROVIDER

##### Customisation d'une DISTRO Yocto FSL Community

- Méthodologie
- Ajout de son propre Layer

##### Travaux pratiques

- Mise en oeuvre de son propre Layer BSP et de sa propre MACHINE
- Génération d'une image optimisée Linux Yocto pour MACHINE dérivée de la SBC Polis i.MX 8M Mini

1.

##### Customisation d'une Image Yocto FSL Community

- Recette minimale de construction de paquetage
- Customisation de recettes
- Ajout de d'étapes de compilation
- Notion de classes
- Syntaxe bitbake avancée

##### Travaux pratiques

- Écriture d'une recette simple Yocto pour l'intégration de votre application à l'image filesystem finale.
- Analyse d'exemples de paquetages et gestion des mises à jour unitaires sous Yocto
- Ajouter des fichiers de configuration à une recette
- Recette d'ajout de script de démarrage avec initd
- Recette d'ajout de service de démarrage sous Systemd

##### Software Development Kit Yocto FSL Community

- Pluggin Eclipse Yocto SDK
- Ajout de package-dev et header kernel pour développement de device driver
- Génération et déploiement d'un SDK complet
- Notion d'Extended-SDK

##### Travaux pratiques

- Mise en oeuvre du SDK GNU GCC/GDB Yocto sous Eclipse : configuration, compilation et débogage sur cible Polis i.MX 8M Mini

3.

##### Customisation d'une DISTRO Yocto FSL Community (suite)

- Bootloader u-boot et barebox
- Spécificités et optimisation du noyau linux-fslc et linux-i.mx sous Yocto
- Customisation du kernel devicetree

##### Travaux pratiques

- Configuration du noyau linux PREEMPT-RT low-latency pour MACHINE dérivée du kit Phytoc phyBOARD Polis i.MX 8M Mini
- Modification du devicetree et mise en oeuvre d'un patch noyau avec l'utilitaire GIT sous Yocto
- Génération d'une image Linux pour SBC Polis i.MX 8M Mini avec partition rootfs readonly

2.

##### Introduction au développement de pilotes Linux / Pilotes de périphériques sous Linux

- Introduction à la programmation en mode noyau
- Architecture d'un module simple
- Programmation de pilote de périphérique
- Programmation de pilotes de périphériques simples : Structure de la File Operation
- API du noyau Linux et gestion mémoire en Kernel Mode

##### Travaux pratiques

- Compilation et déploiement d'un module driver linux externe/propriétaire avec le SDK Yocto LinRT sous Eclipse
- Programmation de pilotes de périphériques Linux, Signaux et Timer en kernel mode sur plateforme i.MX 8
- Driver bloquant, gestion d'interruption sur plateforme sur cible Phytoc Polis i.MX 8M Mini

4.

##### Software Development Kit Qt sous BSP LinRT Yocto FSL Community

- Intégration Qt sous Yocto FSL Community
- Support Hardware GPU OpenGL ES, VPU, Audio
- Support KMS/DRM et support Multi-Framerbuffer
- Spécificités d'un SDK Yocto Qt 5.13 LinRT et support GPU : différents réglages des packages de modules Qt 6.4 entre CPU iMX 8

##### Travaux pratiques

- Personnalisation de l'image fslc-framebuffer avec ajout de layer Qt 6.4 sous Yocto kirkstone 6.4 et des modules QtCANBus, QtSerialPort et QtSerialBus pour le Kit Phytoc Polis i.MX 8M Mini.
- Mise en oeuvre du SDK Qt 6.4 Yocto sous QtCreator : configuration, déploiement et débogage sur cible Polis i.MX 8M Mini (qmake et cmake build).

5.