

Cours de formation, modules condensés pour QNX RTOS

Architecture du système d'exploitation temps réel QNX Neutrino.

Pré-requis : sans

Durée (heure) : 3

Le cours commence par une vue d'ensemble de QNX Neutrino et met en avant les fonctionnalités du micronoyau de QNX Neutrino et de son gestionnaire de processus. Il traite de la protection mémoire et de son adressage, du modèle de processus et des « threads », du séquenceur et de sa gestion du temps. Les différents types de communication interprocessus, leurs synchronisations, la définition du gestionnaire de ressources, la localisation d'objets partagés et du système graphique Photon de QNX pour le multifenêtrage sont aussi abordés dans cette section

Tiré du cours : Développer, déboguer et analyser les performances avec l'Environnement de développement

Environnement de Développement Intégré (IDE), fonctions de base.

Durée (heure) : 2

Donne une vue d'ensemble nécessaire au programmeur qui utilise l'atelier logiciel de développement « QNX Momentics » pour QNX Neutrino et fournit le concept pour naviguer, configurer et personnaliser l'IDE selon les besoins et les préférences. Cette section met en œuvre aussi la gestion de projet, l'espace de travail et les modèles de connexions entre l'IDE et les cartes cibles et introduit les différentes façons de s'interfacer avec les cartes cibles.

Edition et compilation.

Durée (heure) : 1,5

Cette section se focalise sur l'éditeur de code de l'IDE, montrant les fonctions rapides pour aller plus vite. Elle montre aussi comment construire du code et comment l'IDE identifie les problèmes de construction de code.

Exécution et Débogage.

Durée (heure) : 2,5

Cette section, couvre comment lancer des programmes, en mode commande et aussi à partir du débogueur de l'IDE. Elle fournit une vue d'ensemble des fonctionnalités fondamentales telles que sauter dans le code, mettre des points d'arrêts et examiner les données.

Tiré du cours : Développer, déboguer et analyser les performances avec l'Environnement de développement

Synchronisation de processus et de « Threads ».

Durée (heure) : 3,5

Une vue d'ensemble sur les services des Threads dans QNX Neutrino, qui permet l'exécution multiple de programmes. La discussion porte sur ce qu'est un thread, ses priorités, ses algorithmes de séquençement, comment lancer et arrêter un thread, comment synchroniser des accès aux ressources en utilisant les appels de fonctions POSIX de QNX Neutrino. Des exercices permettent de mettre en pratique les concepts clés.

Communication inter-processus (IPC).

Pré-requis : Synchronisation de processus et de « Threads ».

Durée (heure) : 6

Explorer les différentes méthodes que QNX Neutrino propose pour permettre aux processus d'échanger des informations et de les contrôler. L'accent est mis sur le passage de messages qui est un principe fondamentale de l'architecture de QNX Neutrino. Cette section examine aussi les autres méthodes de communication, telles que les impulsions, la mémoire partagée et les messages asynchrones.

Tiré du cours : Programmation temps réel pour le système d'exploitation (RTOS) QNX Neutrino®.

Minuteur, horloge et temps échus.

Pré-requis : Communication inter-processus (IPC).

Durée (heure) : 3,25

Une explication sur comment atteindre les temps du déterminisme demandés avec une discussion sur l'architecture pour la gestion des horloges et du temps, comment gérer une horloge périodique, un click d'horloge, un marquage de temps, les performances et les temps échus.

Tiré du cours : Programmation temps réel pour le système d'exploitation (RTOS) QNX Neutrino®.

Introduction à la programmation du matériel.

Durée (heure) : 1,5

Introduction aux méthodes d'accès du matériel, incluant l'adressage des E/S (entrées/sorties) et de leurs mémoires, la gestion de la DMA et des interruptions.

Introduction aux gestionnaires de ressources

Pré-requis : Communication inter-processus (IPC).

Durée (heure) : 3

Chaque processus peut être un gestionnaire de ressources, et les pilotes en sont généralement. Cette section montre les possibilités des gestionnaires de ressources, incluant les résolutions de nom de chemin, les formats des messages IPC et la structure générale d'un gestionnaire de ressource. Plusieurs exercices pratiques mettent en œuvre comment initialiser un gestionnaire de ressource et gérer les opérations d'écritures et de lectures.

Tiré du cours : Programmation temps réel pour le système d'exploitation (RTOS) QNX Neutrino®.

Profiler une application.

Pré-requis : sans

Durée (heure) : 1

Tous les détails sur comment utiliser l'IDE pour profiler un processus simple. En utilisant le profileur, les parties de votre code qui consomment le plus de temps CPU (processeur) sont identifiées aussi bien au niveau des fonctions qu'au niveau de la ligne de source. Le profileur permet de développer du code de manière plus efficace. L'utilisation au niveau exécution ainsi qu'au niveau post-mortem du profileur d'application est couverte dans cette section.

Couverture de code

Pré-requis : sans

Durée (heure) : 1,5

Déterminer l'efficacité d'une suite de tests nécessite du travail. Cette section montre comment les outils de l'IDE peuvent déterminer combien de code a été réellement exécuté, pointant jusqu'à la ligne du code source par le test.

Tiré du cours : Programmation temps réel pour le système d'exploitation (RTOS) QNX Neutrino®.

Débogue de problèmes mémoire.

Durée (heure) : 2

Une mémoire corrompue peut souvent être corrigés ou évités par des tests exhaustifs ou précis. Cette section explore les outils fournis par l'IDE pour trouver les problèmes de corruption mémoire, les fuites, la consommation de mémoire excessive.

Profiler le Système.

Pré-requis : sans

Durée (heure) : 2,5

Avec le profileur du système, le système complet peut être examiné pour comprendre ce qu'il se passe. Les détails sont disponibles au niveau noyau. La section aborde tout ce qui se passe au niveau noyau ainsi que les étapes nécessaires à la configuration. Elle montre comment récupérer les sessions d'activités ou « log » à travers l'IDE, à partir du mode ligne de commande, ou sous le contrôle d'un programme pour ensuite analyser le résultat des données et les afficher à travers différentes perspectives du profileur. La section se termine par comment insérer ses propres données dans des sessions d'activités.

Tiré du cours : Développer, déboguer et analyser les performances avec l'Environnement de développement

Construction d'une image QNX Neutrino, amorçer un démarrage.

Pré-requis : sans

Durée (heure) : 1

Cette section aborde comment réaliser une image avec les composants du système d'exploitation QNX, tels que le micro-noyau, le réseau, le gestionnaire de fichiers, les pilotes du matériel, les utilitaires et programmes, afin de produire une image du RTOS pour démarrer un système automatiquement. Elle montre aussi les différentes manières d'enregistrer l'image dans le système embarqué.

Tiré du cours : Programmation temps réel pour le système d'exploitation (RTOS) QNX Neutrino®.

Démonstration de fonctions avancées de l'IDE QNX Momentics

les thèmes tels que le profileur d'application, la couverture du code, l'analyse de la mémoire et le profileur système seront abordés et démontrés en fonction du temps restant à la session de formation.