



Windows et les extensions Temps Réel

Source : <https://acontis.com/en/windows-and-realtime.html>

Temps de lecture : 10 min

De plus en plus d'applications, en particulier dans les marchés de l'automatisation industrielle, de la médecine et de l'acquisition et de la mesure de données, nécessitent un comportement temps réel déterministe.

Windows ne prend pas en charge de manière native une opération déterministe en temps réel, mais il existe de nombreuses « extensions temps réel Windows » qui peuvent fournir aux développeurs une solution pour tirer pleinement parti des fonctionnalités de Windows tout en préservant les exigences temps réel de leur application.

Cet article vous propose une série de pistes et de conseils pour mixer temps réel et Windows.

Windows et temps réel, le contexte

La différence entre un système d'exploitation (OS) standard et un OS en temps réel (RTOS) que l'on trouve dans de nombreux systèmes embarqués, est le temps de réponse aux événements externes. Les systèmes d'exploitation standard fournissent généralement une réponse en temps non déterministe, tout en essayant de rester réactifs envers l'utilisateur, parfois en temps réel dit « soft », mais sans garantie quant au moment où chaque tâche sera terminée. Un RTOS diffère car il fournit une réponse en temps réel « dur », fournissant une réaction rapide et hautement déterministe aux événements externes.

Lors de la commutation entre les tâches, le RTOS doit choisir la tâche la plus appropriée à charger ensuite. Il existe plusieurs algorithmes d'ordonnancement disponibles, cependant, pour fournir un système réactif, la plupart des RTOS utilisent un algorithme de planification préemptive, chaque tâche recevant une valeur de priorité individuelle. Plus la réponse requise est rapide, plus le niveau de priorité attribué est élevé, la tâche choisie pour s'exécuter est la tâche de priorité la plus élevée pouvant être exécutée. Il en résulte un système hautement réactif.

On pourrait penser que si vous intégrez votre logiciel dans un pilote Windows, le logiciel peut répondre à des exigences déterministes et temps réel, mais Windows effectuera des demandes au système d'exploitation de manière à obtenir des performances de calcul moyennes maximales et une réponse interactive rapide de l'utilisateur sans réelle garantie de temps de réponse. Il en va de même pour les applications Windows standard, même si les threads s'exécutent avec la priorité temps réel de Windows, ainsi que pour les pilotes Windows, car les mécanismes sous-jacents du système d'exploitation Windows sont les mêmes pour les deux.

Le système d'exécution

Pour garantir des performances déterministes temps réel, le système d'exécution sous-jacent, qui est chargé d'exécuter les opérations temps réel, doit être complètement indépendant de Windows. Il doit fournir une variété de fonctionnalités pour répondre aux exigences temps réel typiques d'aujourd'hui, telles que :

- Gestion de la mémoire
- Multithreading : planificateur déterministe temps réel
- Traitement multiple : exécution de plusieurs instances indépendantes d'applications
- Protection entre la couche application du système d'exécution principal sous-jacent
- Fonctions de synchronisation et de communication des threads / processus
- Prise en charge de plusieurs cœurs (multitraitement symétrique)

Par conséquent, un système d'exploitation temps réel (RTOS) qui fournit au moins certaines des fonctionnalités ci-dessus peut être requis pour le comportement temps réel dans Windows, et ce RTOS doit être complètement indépendant de Windows partout où des exigences temps réel existent.

Isolation

L'isolation est une condition essentielle pour assurer un fonctionnement fiable et sécurisé des applications temps réel aux côtés de Windows. Outre l'isolation du système d'exécution des applications temps réel exécutées, le système d'exécution doit également être isolé et protégé de Windows. La technologie de gestion de la mémoire et de virtualisation doit être utilisée pour obtenir une isolation maximale.

Conservation du temps réel

Même si le système d'exécution utilisé dans une extension temps réel Windows est un RTOS, il existe une variété d'aspects à prendre en compte pour préserver ces fonctionnalités lors de l'exécution avec Windows sur une architecture PC moderne. L'accès aux ressources matérielles partagées, telles que le cache et la RAM, est l'un des aspects les plus critiques. Les interruptions de gestion du système peuvent également entraîner des retards inacceptables. Par conséquent, le système d'exécution doit aborder ces sujets. L'utilisation de la technologie d'allocation de cache permet de surmonter les problèmes de cache partagé tandis que la technologie de virtualisation peut être utilisée pour faire face aux interruptions. En outre, une extension temps réel Windows devrait également fournir des outils pour analyser la réponse temps réel d'un matériel spécifique.

Bibliothèque d'exécution C / C ++

La bibliothèque d'exécution C / C ++ est un ensemble de routines de bas niveau utilisées par le compilateur pour invoquer certains des comportements du système d'exécution, en insérant des appels à la bibliothèque d'exécution dans le binaire exécutable compilé. Le système d'exécution C / C ++ implémente le modèle d'exécution, les fonctions intégrées et d'autres comportements fondamentaux du langage de programmation C / C ++. Outre la prise en charge de base du langage C / C ++, les exigences actuelles vont au-delà et recherchent des bibliothèques de niveau supérieur telles que STL ou Boost.

Environnement de développement

Un environnement de développement intégré (IDE) intègre tous les outils requis tels que le compilateur, l'éditeur de liens et le débogueur dans un seul environnement graphique. Ce qui les rend les plus appréciés des développeurs, c'est leur interface utilisateur et leurs capacités attrayantes. Sous Windows, l'IDE de Microsoft Visual Studio est devenue la norme de facto pour le développement d'applications, et par conséquent, une extension temps réel de Windows doit également pouvoir utiliser cet IDE pour les applications temps réel.

Popularité, flexibilité et portabilité

Pour tirer le meilleur parti du RTOS, l'extension temps réel de Windows doit utiliser un RTOS populaire qui est largement utilisé, avec de nombreuses fonctionnalités, des produits tiers et une documentation disponible, et qui soit flexible dans son utilisation. Si la même application temps réel peut également être exécutée de manière autonome sur un PC sans exécuter Windows, ou même sur un système embarqué basé sur ARM, cela garantira une flexibilité maximale à l'avenir.

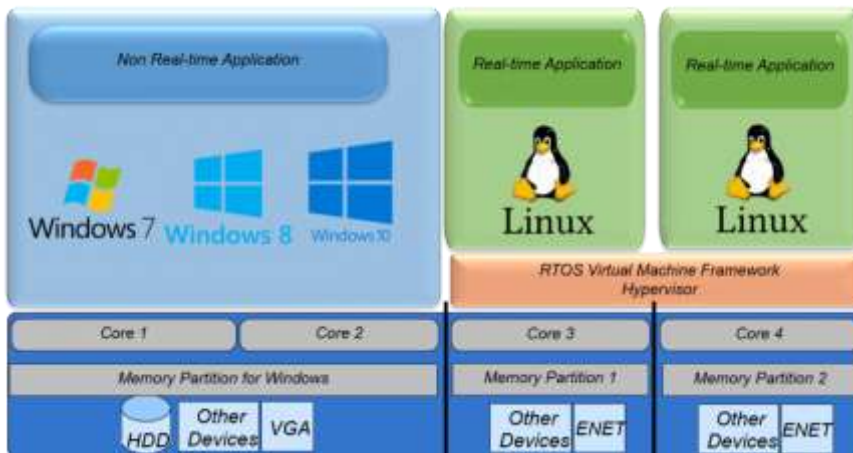
Toutes les extensions temps réel ne sont pas égales

Toutes les extensions temps réel pour Windows ne sont pas implémentées de la même manière, et il y a des avantages et des compromis importants à considérer. Comme décrit ci-dessus et contrairement à ce que les utilisateurs pourraient être amenés à croire, ces extensions temps réel ne font pas de Windows lui-même un RTOS. Les applications Windows existantes ne se comportent pas automatiquement comme des applications temps réel lorsqu'une extension Windows temps réel est implémentée. Au lieu de cela, un système temps réel est nécessaire pour le comportement temps réel - et un tel environnement est un RTOS.

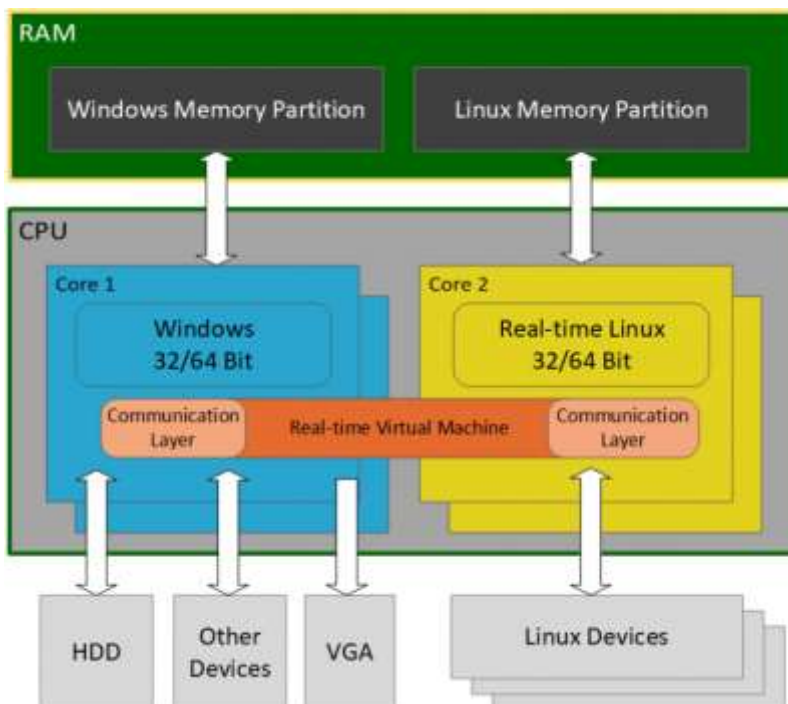
En règle générale, les extensions temps réel de Windows implémentent un RTOS propriétaire en parallèle de Windows, ou parfois même intégré à Windows. Ces solutions RTOS ne sont jamais ou rarement utilisées dans d'autres types d'applications.

Dans ces solutions, toute application qui veut être une application temps réel avec un comportement temps réel doit s'exécuter dans ce RTOS propriétaire. Cela peut être un défi pour les développeurs car ils doivent apprendre et utiliser l'environnement propriétaire pour développer leurs applications temps réel. De plus, ils doivent s'appuyer sur les fonctionnalités qui ont été décidées pour être implémentées par l'éditeur de ce RTOS, qui peuvent parfois faire défaut ou prendre du temps avant que les choses ne soient pleinement disponibles.

LxWin - La solution d'acontis est différente



L'extension temps réel d'acontis, [LxWin](#) fonctionne différemment d'une extension temps réel Windows traditionnelle et offre une alternative intéressante que les utilisateurs et les concepteurs d'applications temps réel devraient envisager. Premièrement, LxWin utilise une approche plus séparée pour les applications temps réel et la partie Windows non temps réel. Plutôt que d'utiliser un RTOS propriétaire, LxWin intègre un noyau temps réel « dur » Linux au-dessus d'un hyperviseur temps réel. Cet hyperviseur est essentiel et garantit une bien meilleure isolation de Windows par rapport aux extensions temps réel traditionnelles.



RT-Linux est le RTOS le plus populaire



De plus, et surtout, LxWin utilise Linux open source pour le RTOS, et non un RTOS propriétaire. C'est une caractéristique clé, car au cours des dernières années, Linux est devenu le système d'exploitation le plus utilisé pour les applications industrielles temps réel. Cela signifie que les développeurs peuvent

utiliser tous les outils et packages populaires déjà disponibles auprès de la communauté open source pour Linux sans avoir besoin d'apprendre un nouveau RTOS propriétaire.

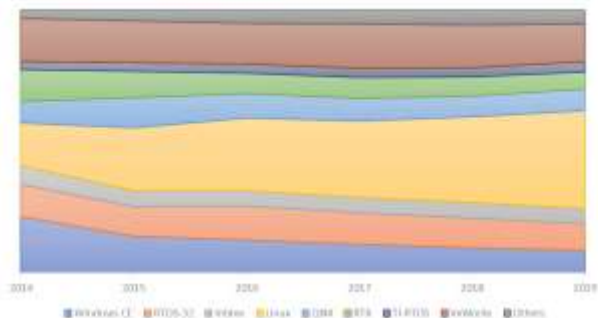
L'utilisation de Linux comme RTOS dans une extension en temps réel Windows est parfaitement logique pour les applications industrielles qui nécessitent un comportement temps réel. À l'origine, l'activité de développement Linux temps réel était dirigée par l'Open Source Automation Development Lab (OSADL), et de grandes entreprises comme AMD, Intel, BMW, Roche, Phoenix Contact et KUKA étaient toutes membres de cette organisation. En 2016, alors que le développement de Linux temps réel devenait de plus en plus populaire, la Linux Foundation a repris la gestion du projet pour pousser le noyau Linux principal lui-même dans un RTOS. Cet objectif devrait enfin être réalisé plus tard cette année ou en 2021. Pour plus d'informations, consultez le site Web du projet Real Time Linux.

acontis est dans une position unique pour voir la tendance des applications industrielles et des systèmes d'exploitation les plus largement utilisés. Notre logiciel EC-Master (pile EtherCAT Master), qui prend aujourd'hui en charge plus de 25 systèmes d'exploitation différents, doit généralement fonctionner dans un système d'exploitation temps réel pour des performances optimales et rapides. Aujourd'hui, nous considérons Linux comme le choix le plus populaire avec plus de 35% de tous les clients EC-Master utilisant Linux, et un nombre croissant de nouveaux clients choisissant Linux, nous nous attendons donc à ce que ce pourcentage augmente.

acontis EtherCAT Master Real-time OS Adoption :



acontis EtherCAT Master Real-time OS Adoption over Time



Nous voyons également cette tendance sur le marché industriel plus large, au-delà des seules applications EtherCAT. Les principales entreprises mondiales fournissant des contrôleurs de machine, de mouvement et de robotique, ainsi que des API, des machines de fabrication de semi-conducteurs et des solutions de test et de mesure, ont toutes commencé à mettre en œuvre Linux-RT au lieu d'autres systèmes d'exploitation en temps réel hérités.

Enfin, dans le prochain standard Ethernet temps réel connu sous le nom de Time Sensitive Networking (TSN), RT-Linux jouera un rôle clé en tant que système d'exploitation le plus favorisé.

Outre RT-Linux, acontis fournit des solutions alternatives pour des cas d'utilisation spécifiques, où par exemple le client utilise déjà un autre RTOS de manière native.

Comment RT-Linux satisfait toutes les exigences

En tenant compte des exigences susmentionnées pour une extension temps réel Windows, il est évident que RT-Linux est le meilleur choix d'aujourd'hui.

Le système d'exécution

RT-Linux fournit toutes les fonctionnalités pour répondre aux exigences actuelles d'un RTOS. C'est l'un des systèmes d'exploitation les plus puissants, sinon le plus puissant du marché. En fonction des besoins réels, il peut également être réduit de manière significative pour n'inclure que les composants typiques nécessaires pour les applications temps réel. Linux n'est pas seulement utilisé dans les serveurs, mais aussi dans les petits appareils embarqués comme les montres intelligentes ou les téléviseurs où l'empreinte mémoire est toujours un challenge. Ainsi, RT-Linux est également petit et puissant.

Pour les applications exigeantes en CPU, LxWin prend également en charge le multiprocesseur symétrique où plusieurs cœurs de processeurs peuvent être utilisés pour l'application temps réel.

Isolation

RT-Linux lui-même a une isolation sécurisée et fiable entre les applications temps réel et le noyau Linux. Bien que cela ne soit pas critique pour les applications temps réel, il corrige même des vulnérabilités telles que Meltdown.

De plus, Windows et RT-Linux sont isolés de manière sécurisée à l'aide de mécanismes MMU, une isolation encore plus grande peut être obtenue en activant la technologie de virtualisation Intel

Conservation du temps réel

LxWin fournit un outil simple pour analyser les capacités temps réel. 3 voyants indiquent rapidement si le système est suffisant pour les applications temps réel. acontis a récemment commencé à créer une sélection de matériels pré-validés pouvant être utilisés pour des applications temps réel. Pour améliorer les capacités temps réel, CAT ainsi qu'Intel VT peuvent être utilisés

Bibliothèque d'exécution C / C ++, pilotes, logiciels tiers

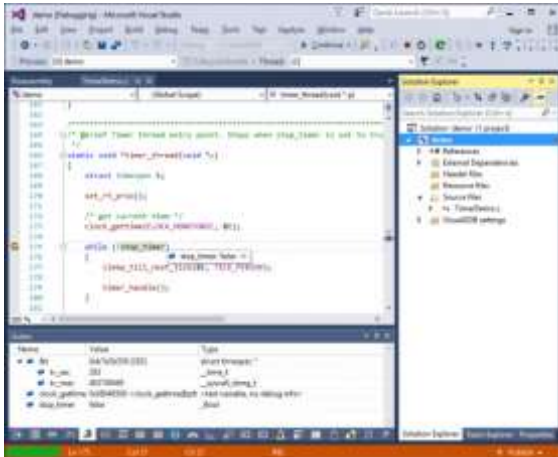
En plus de pouvoir être un noyau temps réel, Linux lui-même apporte de nombreux avantages utiles. Des fonctionnalités sophistiquées telles que les sockets, une API POSIX complète, l'isolation de l'espace utilisateur / espace noyau, le multitâche et le multitraitements sont tous disponibles. Pour presque tous les matériels disponibles sur le marché aujourd'hui, un pilote Linux est disponible. Cela signifie que les développeurs n'ont pas besoin d'attendre qu'un pilote spécifique soit implémenté et publié par le fabricant d'une plate-forme d'extension temps réel propriétaire. Enfin, avec Linux, il existe une multitude de bibliothèques de logiciels, de piles de protocoles et de middlewares (produits commerciaux open source et open source) : C++ STL, Boost, Corba, OPC / UA, DDS, AVB / TSN, EtherCAT, PROFINET, API basé sur logiciel, bibliothèques de contrôle de mouvement, solutions CNC et bien d'autres.

Pour les clients utilisant la technologie EtherCAT, acontis fournit la plate-forme EC-Win Windows EtherCAT basée sur LxWin. Enfin, les clients ayant besoin de simuler un réseau EtherCAT en temps réel peuvent exécuter le logiciel de simulation acontis EtherCAT EC-Simulator sur Windows en utilisant LxWin.

Environnement de développement

La partie principale du développement et du débogage des applications temps réel est la prise en charge de Microsoft Visual Studio. Les assistants de projet permettent aux clients d'obtenir des applications temps réel créées de manière simple et logique - aussi simple que de créer des projets d'application Windows.

Le débogage est également effectué de la même manière que le débogage des applications Windows classiques.



Alternativement, les clients familiers avec Eclipse peuvent également utiliser ce puissant IDE, sous Linux ou sous Windows.

Popularité, flexibilité et portabilité

Linux est aujourd'hui le système d'exploitation le plus populaire pour les systèmes embarqués. Il est disponible sur Intel x86 (32 bits et 64 bits). Les applications écrites pour la plate-forme PC peuvent facilement être portées vers l'architecture ARM intégrée car le système d'exploitation garantit la réutilisabilité des sources existantes.

Les clients utilisant Microsoft Windows Embedded Compact (anciennement Windows CE) doivent actuellement passer à une autre solution, car Microsoft cessera de fournir des mises à jour et une assistance pour ces systèmes. Le chemin de migration le plus populaire est RT-Linux et on peut envisager d'utiliser LxWin si les fonctionnalités Windows doivent être préservées.

Les défis de la création d'un noyau temps réel Linux

Il y a, bien sûr, encore quelques défis à utiliser Linux pour les applications temps réel. L'une des tâches les plus difficiles consiste à configurer le noyau Linux pour un comportement temps réel. Jusqu'à ce que le noyau Linux principal soit entièrement réalisé en tant que noyau temps réel, les développeurs doivent aujourd'hui patcher le noyau et configurer le noyau spécialement pour atteindre le temps réel. Ceci est souvent appelé le patch RT-PREEMPT, mais il peut s'agir de nombreux patches et tous ne fonctionnent pas parfaitement.

Cependant, avec LxWin, les développeurs n'ont pas du tout à se soucier de la configuration et de la compilation du noyau Linux. Le noyau Linux fourni avec LxWin est déjà correctement patché, configuré et testé en tant que noyau temps réel « dur » avec un comportement déterministe. En conséquence, des temps de cycle extrêmement courts jusqu'à 50 microsecondes peuvent être atteints.

Résumé des fonctionnalités les plus importantes de LxWin

- Disponible pour Windows 7 et versions ultérieures de Windows (32 bits et 64 bits)
- Temps de réponse rapide en quelques microsecondes et indépendance totale de Windows, si Windows tombe en panne, la partie temps réel continue de fonctionner !
- Utilisation optimisée des systèmes multicœurs (évolutif d'un à plusieurs cœurs CPU pour la partie temps réel)
- API temps réel intuitive et populaire (processus, threads, événements, sockets, support POSIX, ...)
- Prise en charge des outils de développement Microsoft (Visual Studio)
- Environnement de débogage confortable pour la partie temps réel à l'aide des outils Microsoft, les applications en temps réel seront développées et déboguées comme une application Windows normale

Source : <https://acontis.com/en/windows-and-realtime.html>