

Université d'Ottawa
Faculté de génie

École de science informatique
et de génie électrique



University of Ottawa
Faculty of Engineering

School of Electrical Engineering
and Computer Science

CEG4166/CSI4141/SEG4145 Conception de systèmes en temps réel

Hiver 2024

Professeur : Mohamed Ali Ibrahim, Ph.D., P. Eng.

Comprendre configUSE_TIME_SLICING dans FreeRTOS

Source : <https://www.freertos.org/>

Plan

- Qu'est-ce que le découpage (slicing) temporel ?
- Comment fonctionne le découpage temporel dans un système multitâche préemptif ?

Définition du découpage temporel

Le découpage temporel est une méthode utilisée dans les systèmes d'exploitation multitâches préemptifs pour allouer une quantité fixe de temps CPU, appelée "tranche de temps" ou "quantum", à chaque processus ou thread dans une file d'attente de tâches exécutables.

Les points clés du découpage du temps

- **Attribution équitable du temps d'utilisation de l'unité centrale** : le découpage du temps garantit que chaque processus ou thread a la même possibilité d'utiliser l'unité centrale, ce qui empêche un processus unique de monopoliser les ressources du processeur.
- **Ordonnancement préemptif** : Dans l'ordonnancement temporel, le planificateur du système d'exploitation interrompt de force un processus en cours d'exécution après l'expiration de la tranche de temps qui lui est allouée et passe le contrôle de l'unité centrale au processus suivant dans la file d'attente. On parle d'ordonnancement préemptif car le système d'exploitation préempte (interrompt) le processus en cours d'exécution pour passer à un autre processus.
- **Changement de contexte** : lorsque l'ordonnanceur passe d'un processus à un autre, un changement de contexte se produit. Cela implique la sauvegarde de l'état du processus en cours d'exécution et le chargement de l'état du processus suivant à exécuter.

Les points clés du découpage du temps

- **Amélioration de la réactivité** : Le découpage temporel peut améliorer la réactivité d'un système en permettant à plusieurs processus de progresser au cours d'une période donnée, donnant ainsi l'illusion d'une exécution simultanée, en particulier dans les systèmes dotés d'un seul cœur d'unité centrale.
- **Prise en compte de la longueur du quantum** : La longueur de la tranche de temps (quantum) est cruciale ; si elle est trop courte, le système risque de passer trop de temps à changer de contexte, ce qui le rendra inefficace. Si elle est trop longue, elle peut entraîner une mauvaise réactivité, car les tâches peuvent attendre trop longtemps leur tour d'exécution.

Comment fonctionne le découpage temporel dans un système multitâche préemptif ?

- **Priorité des tâches** : Dans un environnement multitâche, chaque processus ou thread a un niveau de priorité associé. Le planificateur utilise ces priorités pour déterminer l'ordre d'exécution des tâches, les tâches les plus prioritaires étant généralement exécutées avant les moins prioritaires.
- **Allocation quantique** : Le système alloue un intervalle de temps fixe, appelé quantum ou tranche de temps, à chaque tâche. Ce quantum est la durée maximale d'exécution continue d'une tâche avant qu'elle ne soit préemptée.
- **Exécution de la tâche** : L'ordonnanceur sélectionne la tâche la plus prioritaire qui est prête à être exécutée et lui alloue l'unité centrale. La tâche s'exécute ensuite pendant la durée qui lui est allouée.

Comment fonctionne le découpage temporel dans un système multitâche préemptif ?

- **Préemption et changement de contexte** : Si la tâche ne s'achève pas dans sa tranche de temps, l'ordonnanceur la préempte (c'est-à-dire qu'il interrompt de force son exécution) une fois que son quantum a expiré. Le système enregistre alors l'état actuel de la tâche (contexte) afin qu'elle puisse être reprise ultérieurement. Cet état sauvegardé comprend le compteur de programme de la tâche, les valeurs des registres et d'autres données pertinentes.
- **Changement de contexte** : le planificateur sélectionne la tâche suivante à exécuter, qui peut être une autre tâche de même priorité (en mode round-robin) ou une tâche de priorité plus élevée qui est devenue prête à être exécutée. Le système effectue alors un changement de contexte, en chargeant l'état sauvegardé de la nouvelle tâche pour reprendre ou démarrer son exécution.
- **Continuation du cycle** : Ce cycle d'exécution, de préemption et de changement de contexte se poursuit, ce qui permet à toutes les tâches prêtes d'obtenir une part équitable de l'unité centrale au fil du temps, facilitant ainsi le multitâche.

Comment fonctionne le découpage temporel dans un système multitâche préemptif ?

- **Achèvement ou blocage de la tâche** : Si une tâche termine son exécution ou se bloque (par exemple, en attente d'E/S ou d'une primitive de synchronisation), elle renonce volontairement à l'unité centrale avant l'expiration de son quantum. L'ordonnanceur sélectionne alors la tâche suivante à exécuter en fonction des priorités actuelles et de l'état de préparation des tâches.
- **Ajustement du quantum** : Certains systèmes ajustent dynamiquement la longueur du quantum pour chaque tâche en fonction de son comportement et de ses exigences, afin d'optimiser la réactivité et l'efficacité du système.

Le paramètre de configuration configUSE_TIME_SLICING dans FreeRTOS

- Le paramètre de configuration configUSE_TIME_SLICING de FreeRTOS contrôle si le découpage temporel est activé ou désactivé pour les tâches qui ont le même niveau de priorité.
- Ce paramètre joue un rôle crucial dans la manière dont le planificateur alloue le temps de l'unité centrale à des tâches de priorité égale au sein du système d'exploitation.

Qu'est-ce que configUSE_TIME_SLICING ?

- **Objectif** : configUSE_TIME_SLICING détermine si le noyau FreeRTOS utilisera le découpage temporel (round-robin scheduling) pour les tâches qui ont le même niveau de priorité.
- **Type** : Il est généralement défini comme une macro du préprocesseur qui peut prendre la valeur 1 (activé) ou 0 (désactivé).

Lorsque le découpage temporel est activé (configUSE_TIME_SLICING = 1)

- **Comportement** : Si le découpage temporel est activé, l'ordonnanceur FreeRTOS s'assure que les tâches de même niveau de priorité partagent le temps CPU de manière égale. Lorsque plusieurs tâches de même priorité sont prêtes à s'exécuter, chaque tâche est autorisée à s'exécuter pendant une tranche de temps fixe ou un quantum.
- **Avantage** : Cela permet d'éviter qu'une tâche unique ne monopolise l'unité centrale et de répartir équitablement le temps d'utilisation de l'unité centrale entre des tâches de priorité égale, ce qui est particulièrement important dans les systèmes nécessitant un fonctionnement multitâche réactif.

Lorsque le découpage temporel est désactivé (configUSE_TIME_SLICING = 0)

- **Comportement** : La désactivation du découpage temporel signifie que lorsqu'une tâche commence à s'exécuter, elle continue à le faire jusqu'à ce qu'elle se bloque, cède sa place ou soit préemptée par une tâche de priorité supérieure. Les autres tâches de même niveau de priorité ne seront pas prises en compte pour l'exécution jusqu'à ce que la tâche en cours abandonne volontairement l'unité centrale.
- **Avantage** : Cela peut être utile dans des scénarios où des tâches de même priorité ne requièrent pas nécessairement le même temps d'utilisation de l'unité centrale ou lorsqu'il est souhaitable de réduire la surcharge de la commutation de contexte pour améliorer l'efficacité du système.

Configuration et utilisation

- **Définition du paramètre** : `configUSE_TIME_SLICING` est typiquement défini dans le fichier `FreeRTOSConfig.h`, qui contient les constantes de configuration du noyau FreeRTOS.
- **Personnalisation** : Les développeurs peuvent personnaliser ce paramètre en fonction des exigences de l'application. Par exemple, dans un système où des tâches de même niveau de priorité doivent être exécutées pendant des durées différentes ou dans des applications où il est essentiel de minimiser le changement de contexte, la désactivation du découpage temporel peut s'avérer bénéfique.

Impact sur les performances du système

- Désactiver le découpage temporel dans FreeRTOS (en mettant `configUSE_TIME_SLICING` à 0) peut être bénéfique dans certains scénarios où les caractéristiques des tâches ou les exigences de l'application dictent que les tâches doivent s'exécuter sans interruption aussi longtemps que possible, ou lorsque la minimisation du changement de contexte est une priorité.

Tâches de calcul à haute performance

- **Scénario** : dans les applications nécessitant des tâches de calcul intensives qui doivent fonctionner aussi efficacement que possible sans interruptions fréquentes.
- **Avantage** : la désactivation du découpage temporel permet à ces tâches d'utiliser pleinement l'unité centrale jusqu'à ce qu'elles aient achevé une grande partie de leur travail ou atteint un point de blocage naturel, ce qui réduit la surcharge causée par le changement de contexte.

Systèmes en temps réel à comportement déterministe

- **Scénario** : dans les systèmes en temps réel où les tâches ont des délais d'exécution strictement définis et où le déterminisme est essentiel.
- **Avantage** : la désactivation du découpage temporel garantit qu'une fois qu'une tâche prioritaire commence à s'exécuter, elle peut continuer sans préemption due au découpage temporel, ce qui permet d'obtenir des temps d'exécution des tâches plus prévisibles et déterministes.

Tâche avec de longues opérations non préemptives

- **Scénario** : Lorsque des tâches effectuent des opérations qui ne doivent pas être interrompues, telles qu'une séquence d'opérations matérielles étroitement liées ou des sections critiques de code qui nécessitent une exécution atomique.
- **Avantage** : le découpage temporel étant désactivé, ces tâches peuvent achever leurs opérations critiques sans risquer d'être devancées, ce qui préserve l'intégrité de l'opération.

Réduction de la charge de travail liée au changement de contexte

- **Scénario** : dans les systèmes où la surcharge liée au changement de contexte est un problème, en particulier dans les environnements à faible consommation d'énergie et à ressources limitées.
- **Avantage** : la désactivation du découpage temporel minimise le nombre de changements de contexte, ce qui permet d'économiser des cycles de l'unité centrale et de l'énergie, ce qui peut s'avérer crucial pour prolonger la durée de vie de la batterie des appareils portables.

Réduction de la charge de travail liée au changement de contexte

- **Scénario** : Lorsqu'une tâche d'arrière-plan doit être exécutée en permanence mais doit céder la place à des tâches plus prioritaires lorsqu'elles sont prêtes.
- **Avantage** : la tâche d'arrière-plan peut monopoliser l'unité centrale lorsqu'aucune tâche prioritaire ne doit être exécutée, ce qui garantit une utilisation efficace du processeur tout en permettant aux tâches prioritaires de prendre le relais en cas de besoin.

Réduction de la charge de travail liée au changement de contexte

- Lorsque vous décidez d'activer ou de désactiver `configUSE_TIME_SLICING` sous FreeRTOS, il est important de prendre en compte les besoins spécifiques de votre application et la façon dont les tâches interagissent entre elles.

Comprendre les priorités des tâches

- **Relations entre les tâches** : Soyez clair sur la manière dont les tâches sont liées les unes aux autres et sur leurs niveaux de priorité. Le découpage du temps affecte principalement les tâches de même priorité.
- **Conception des priorités** : Définissez soigneusement les priorités de vos tâches afin de vous assurer que les tâches critiques ont des priorités plus élevées et sont moins affectées par le découpage temporel.

Analyse du comportement des tâches

- **Durée de la tâche** : Considérez la durée d'exécution typique des tâches. Les tâches de longue durée peuvent bénéficier d'un découpage temporel désactivé, tandis que les tâches courtes et interactives peuvent bénéficier d'un découpage temporel activé.
- **Blocage et abandon** : Comprendre la fréquence à laquelle les tâches se bloquent ou cèdent. Si les tâches attendent fréquemment des primitives d'E/S ou de synchronisation, l'impact du découpage temporel peut être réduit.

Performance du système

- **Frais généraux liés à la commutation de contexte** : Évaluez l'impact de la commutation de contexte sur les performances de votre système. La désactivation du découpage temporel peut réduire les changements de contexte, mais peut entraîner une sous-utilisation de l'unité centrale si elle n'est pas gérée correctement.
- **Utilisation de l'unité centrale** : Surveillez l'utilisation du processeur. La désactivation du découpage temporel peut entraîner une utilisation inefficace de l'unité centrale si toutes les tâches n'ont pas suffisamment de travail pour occuper l'unité centrale.

Réactivité et équité

- **Réactivité du système** : Veiller à ce que le système réponde à ses exigences en matière de réactivité. Le découpage temporel peut contribuer à rendre le système plus réactif en permettant aux tâches de partager équitablement le temps de l'unité centrale.
- **L'équité** : Dans un système où les tâches ont la même importance et doivent progresser à un rythme similaire, la possibilité d'un découpage temporel garantit l'équité.

Essais et optimisation

- **Simulation et test** : Testez les performances du système avec les deux paramètres de `configUSE_TIME_SLICING`. Simulez différentes conditions de charge pour en comprendre l'impact.
- **Ajustement dynamique** : Examinez si votre système peut bénéficier d'un ajustement dynamique du découpage temporel ou de l'utilisation d'autres fonctions de planification avancées en fonction de l'état actuel du système ou de la charge.

Documentation et maintenance

- **Documentation de la configuration** : Documentez les raisons de l'activation ou de la désactivation du découpage temporel dans la configuration de votre système afin de faciliter la maintenance et le débogage futurs.
- **Révisions du code** : Examinez régulièrement les priorités des tâches et la logique de planification dans le cadre de la maintenance de votre code pour vous assurer qu'elles correspondent aux exigences du système et aux attentes en matière de performances.

Conclusion

- La décision d'activer ou de désactiver `configUSE_TIME_SLICING` doit être basée sur une compréhension approfondie des exigences de votre application, des comportements des tâches et des objectifs de performance du système.
- Des tests réguliers et une analyse des performances sont essentiels pour trouver la configuration optimale pour votre scénario spécifique.