



## Microcontrôleur, source d'horloge et DMA

### Choix d'un microcontrôleur

Références considérées

- nRF52840 de Nordic Semiconductor
- STM32WB55CG de STMicroelectronics
- EFR32BG12 de Silicon Labs

Ci-dessous un tableau de comparaison des principales caractéristiques.

Reference	nRF52840	STM32WB55CG	EFR32BG12
Fabricant	Nordic Semiconductor	STMicroelectronics	Silicon Labs
Package	aQFN73 48 pins, 7 x 7 mm WLCSP93 48 pins, 3.5x3.6mm	UFQFPN 48 pins 7x7mm VFQFPN68 8 x 8 mm WLCSP100 4.8 x 5.1 mm BGA129	QFN32 4x4 mm Package
Cœur	Cortex M4 with FPU @ 64MHz	Cortex M4 with FPU @ 64MHz	Cortex M33 with FPU @ 80MHz
Architecture mémoire	Flash: 1 MB RAM: 256kB	Flash: up to 1MB RAM: up to 256KB	Flash: up to 1MB RAM: up to 96KB
Fonctionnement Radio	Stack Bluetooth intégrée a la flash 152kB	Dedicated 32-bit Cortex M0 + CPU for real-time Radio layer	Stack Bluetooth intégrée a la flash 80-100kB
Consommation sleep	3.16 µA @ 3 V in OFF mode + full RAM Retention	2.1µA @3V in stop mode + full RAM retention +RTC	4.5 µA @ 3V in deep sleep + full ram retention
TX current	4.8 mA @ 0 dBm with DC/DC	5.2 mA @ 0dBm	8.8mA @ 0dBm
RX current	4.6 mA @ 1 Mbps	4.5 mA @ 1Mbps Avec assez peu de détail	9.3 mA @ 1Mbps
Périphérique	USB 2.0, 2 UART, 4 SPI, QSPI, 2 I2C, ADC@12bit	USB2.0, 2 UART, 2 SPI, QSPI, 2 I2C, ADC@12bit	3 UART, 3 SPI, 2 I2C, ADC@12bit
Puissance max de sortie	TX Power -20 to +8 dBm in 4 dB steps	TX power up to +6 dBm with 1 dB steps	Up to +19dBm
Receiver sensitivity Bluetooth	95 dBm @ 1Mbps	96 dBm @ 1Mbps	97.5 dBm @ 1Mbps
Supply voltage	Range 1.7 V to 5.5 V	Range 1.71 to 3.6	Range 1.71 V to 3.8 V
Prix	3.48€ @ 1000	3.92€ @ 1000	3.05€ @ 1000



En reprenant les différents paramètres, on peut faire les commentaires suivants :

<i>Reference</i>	<b>Commentaire</b>
<i>Package</i>	Pour le package, les trois fabricants proposent une version QFN équivalente. Pour les applications fortement contraintes en place, seul le STM32WB55CG existe en version BGA.
<i>Cœur</i>	Les cœurs sont semblables.
<i>Architecture mémoire</i>	Les trois microcontrôleurs sont équivalents en terme de taille de mémoire flash. L'EFR32BG12 a une taille maximale de RAM plus faible, ce qui peut être limitant dans le cas d'utilisation du BLE. En effet, la stack en consomme une partie fixe non négligeable (~106kB) qui est augmentée à chaque connexion. Cela peut aussi être limitant pour tout ce qui est usage de buffers de taille conséquente (réception donnée...).
<i>TX current</i> <i>RX current</i>	Le produit est en advertising constant pour signaler sa présence. Les TX et RX courants sont donc une composante majeure pour optimiser la durée de vie du produit. Le STM32WB55CG et le nRF52840 sont légèrement plus performants sur cet aspect-là.
<i>Consommation sleep</i>	Les consommations en sleep sont assez similaires sur les 3 microcontrôleurs avec un léger désavantage pour le EFR32BG12.
<i>Périphérique</i>	Les périphériques disponibles sont assez proches. Cependant le EFR32BG12 ne dispose pas d'USB ni de QSPI. Nous envisageons d'utiliser une mémoire flash en QSPI ce qui peut être un frein pour l'utilisation de ce microcontrôleur.
<i>Puissance max de sortie</i>	Le EFR32BG12 a une puissance maximum de 19dBm vis-à-vis des 6 et 8dBm de la concurrence.
<i>receiver sensitivity</i>	Le EFR32BG12 est légèrement meilleur.
<i>Supply voltage</i>	La plage de consommation est plus étendue sur le nRF52840 ce qui peut être un avantage pour les applications ayant des périphériques dont l'alimentation minimale est de 5V pour éviter de devoir ajouter un étage de boost.
<i>Prix</i>	Le prix par 1000 des trois références est assez proche. Pour les gros volumes le EFR32BG12 est le moins cher, vient ensuite le nRF52840 puis le STM32WB55CG.

La comparaison des trois références montre que chaque composant a des avantages sur les autres. C'est la mise en parallèle de ces avantages/inconvénients par rapport au besoin de l'application qui permettra de choisir.

Le EFR32BG21 est le plus performant en puissance d'émission radio et est plus orienté vers des applications où une grande portée est recherchée.

Le STM32WB55CC possède un microcontrôleur dédié à la radio, ainsi que de multiples interfaces. Il semble plus adapté pour des applications avec de multiples périphériques où le partage du cœur applicatif est moins facile.



Le nrf52840 est un bon intermédiaire : il possède de nombreuses fonctions mais est moins performant en radio et n'a pas de cœur dédié à la stack BLE. Il se consacre plus à des applications de taille moyenne avec une utilisation de la radio raisonnable.

## Choix de l'horloge

Un microcontrôleur peut utiliser une horloge interne (généralement un oscillateur RC) ou une horloge externe (généralement un quartz ou un oscillateur à cristal) pour déterminer le rythme de son fonctionnement. Le choix entre ces deux options dépend de plusieurs facteurs, et il y a des compromis associés à chaque option. Voici quelques-uns de ces compromis :

1. **Précision** : Les oscillateurs à quartz offrent une précision supérieure aux oscillateurs RC. En raison de la nature physique du cristal de quartz, ils ont une très faible dérive de fréquence en fonction de la température et du temps. Les oscillateurs RC, en revanche, peuvent être affectés par des variations de température et de tension, ce qui peut entraîner des variations de la fréquence d'horloge.
2. **Coût** : Les oscillateurs RC sont moins chers que les oscillateurs à quartz puisque intégrés directement dans le microcontrôleur. En outre, ils ne nécessitent pas de composants externes, ce qui peut réduire encore davantage le coût total du système. Les quartz, en revanche, sont plus coûteux et nécessitent des condensateurs externes pour fonctionner correctement.
3. **Consommation d'énergie** : Les oscillateurs à quartz consomment généralement moins d'énergie que les oscillateurs RC. Si l'économie d'énergie est une considération importante pour votre application, un oscillateur à quartz pourrait être une meilleure option.
4. **Encombrement** : Les oscillateurs RC sont intégrés dans le microcontrôleur et n'occupent donc pas d'espace supplémentaire sur la carte de circuit imprimé (PCB). Les quartz, en revanche, sont des composants externes et nécessitent de l'espace supplémentaire sur la carte, ce qui peut être un facteur limitant pour les designs compacts.
5. **Flexibilité** : Certains microcontrôleurs permettent de modifier la fréquence d'un oscillateur RC par programmation, ce qui peut être utile pour les applications qui nécessitent différents rythmes de fonctionnement à différents moments. Les quartz ont généralement une fréquence fixe qui ne peut pas être modifiée sans remplacer le composant.
6. **Fiabilité** : Les quartz sont généralement plus fiables sur la durée. Les oscillateurs RC peuvent subir des dérives de fréquence à long terme qui peuvent affecter la fiabilité du système.

En résumé, le choix entre un oscillateur RC interne et un quartz externe dépend de vos priorités en termes de coût, de précision, de consommation d'énergie, d'encombrement, de flexibilité et de fiabilité.



## Le Direct Memory Access DMA

Le DMA, ou Direct Memory Access, est une technique utilisée dans les microcontrôleurs pour permettre à certains périphériques de transférer des données directement vers et depuis la mémoire sans solliciter le microcontrôleur lui-même.

Lorsqu'un périphérique, tel qu'une interface série, un convertisseur analogique-numérique (CAN) ou un module de capture d'événements, doit transférer une grande quantité de données vers ou depuis la mémoire du microcontrôleur, le DMA peut être utilisé. Il permet au périphérique de contourner le microcontrôleur et de communiquer directement avec la mémoire, ce qui accélère les transferts de données et libère le microcontrôleur pour effectuer d'autres tâches.

Le fonctionnement du DMA dans un microcontrôleur est généralement le suivant :

1. Configuration : Le microcontrôleur est programmé pour configurer le contrôleur DMA en spécifiant les adresses de début et de fin de la zone mémoire où les données seront transférées, ainsi que le sens du transfert (lecture ou écriture).
2. Demande de transfert : Le périphérique demande l'accès au bus mémoire et au contrôleur DMA pour effectuer le transfert. Le microcontrôleur peut être notifié de cette demande via une interruption ou en vérifiant régulièrement l'état du contrôleur DMA.
3. Transfert direct : Une fois que le périphérique obtient l'accès au bus mémoire, il peut transférer les données directement vers ou depuis la mémoire sans passer par le microcontrôleur. Le contrôleur DMA se charge de l'acheminement des données et de la gestion des adresses.
4. Notification de fin : Une fois le transfert terminé, le contrôleur DMA peut notifier le microcontrôleur de sa complétion via une interruption ou d'autres mécanismes de signalisation.

Le DMA dans les microcontrôleurs offre des avantages similaires à ceux des ordinateurs, tels que :

- Amélioration des performances : En permettant aux périphériques d'accéder directement à la mémoire, le DMA réduit la charge sur le microcontrôleur et accélère les transferts de données, améliorant ainsi les performances globales du système.
- Libération du microcontrôleur : En évitant au microcontrôleur de gérer les transferts de données, le DMA libère sa capacité de traitement pour d'autres tâches, ce qui est particulièrement précieux dans les systèmes embarqués où les ressources sont limitées.
- Transferts asynchrones : Le DMA permet des transferts de données asynchrones, ce qui signifie que le périphérique et le microcontrôleur peuvent fonctionner de manière indépendante. Cela facilite la réalisation de tâches simultanées et peut améliorer la réactivité du système.

Il convient également de prendre en compte les défis et risques potentiels liés à l'utilisation du DMA dans les microcontrôleurs, tels que la gestion des accès concurrents à la mémoire et la nécessité de mettre en place des mécanismes de protection appropriés pour garantir l'intégrité des données.



En résumé, le DMA dans les microcontrôleurs est une technique qui permet à certains périphériques de transférer directement des données