

Déployer des microcontrôleurs conçus pour des tâches spécifiques simplifie les systèmes complexes

En déchargeant le microcontrôleur ou microprocesseur principal de certaines tâches et actions, les microcontrôleurs conçus pour des tâches spécifiques aident à simplifier le processus de conception dans de nombreuses applications. Explications de Microchip.

Dans le monde de l'électronique que l'on connaît aujourd'hui, l'usage de gros microcontrôleurs et microprocesseurs 32 bits faisant tourner un système d'exploitation temps réel RTOS (Real Time Operating System) devient de plus en plus courant. Néanmoins, un seul gros microcontrôleur exécutant une application complexe peut rencontrer des problèmes au niveau des ressources CPU au regard de petites tâches récurrentes qui ne sont pas complexes mais qui s'avèrent chronophages. Dans ce cadre, des composants plus petits, comme les microcontrôleurs 8 bits et 16 bits, peuvent être utilisés pour décharger la puce 32 bits de ces tâches. De quoi laisser suffisamment d'espace à l'innovation et aux possibilités de développement grâce au large éventail de microcontrôleurs disponibles, du 8 bits au 32 bits.

De nombreuses tâches sont en effet plus faciles à gérer si le processeur 32 bits en délègue certaines sur des « sous-processeurs » qui prennent alors à leur charge uniquement une ou deux de ses tâches, tout en ne requérant qu'un minimum de supervision pendant l'exécution. Les ressources CPU sont ainsi libérées au niveau du processeur principal, ce qui réduit la complexité logicielle tout en améliorant les performances et le temps d'exécution.

Par exemple, un microcontrôleur 8 bits peut être utilisé pour créer un module d'extension d'E/S. Les modules d'extension d'E/S ne sont pas des composants complexes,

AUTEUR



Robert Perkel, ingénieur d'application, Microchip Technology.

mais ils consomment beaucoup de temps CPU à cause de la fréquence des interruptions qu'il est nécessaire de gérer. En utilisant un microcontrôleur spécifique pour cette tâche, le composant le plus gros peut réduire le temps et les ressources consacrées aux E/S ainsi que le nombre d'interruptions qu'il doit gérer. De plus, les fonctionnalités du module d'extension d'E/S sont définies au niveau logiciel, ce qui leur permet d'être personnalisées et affinées selon les besoins de l'application.

L'avantage principal de créer un module d'extension d'E/S à l'aide d'un microcontrôleur 8 bits réside dans sa flexibilité. En effet, les fonctionnalités d'un circuit ASIC utilisé pour gérer les extensions d'E/S sont intégrées au sein même du composant, tandis que le microcontrôleur, lui, définit son comportement en se basant sur le logiciel qu'il exécute. Grâce à cette flexibilité, la version reposant sur un microcontrôleur peut être adaptée aux besoins de l'application finale.

Mise en œuvre d'un module d'extension d'E/S avancé

Au niveau interne, le module d'extension d'E/S avancé s'appuie sur une structure reposant sur une table de correspondances (LUT). Avant toute opération de lecture ou écriture, une adresse virtuelle est envoyée. Cette adresse ne correspond pas aux registres du microcontrôleur; elle renvoie uniquement à la table de correspondances. Ainsi, les fonctionnalités absentes des registres matériels du microcontrô-

leur peuvent être ajoutées de façon transparente. Les entrées dans la table de correspondances peuvent également être facilement réorganisées pour différents usages. Autre avantage de cette structure: la possibilité d'ajouter des permissions à la table de correspondances. Par exemple, pour créer un registre en lecture seule, il suffit d'omettre l'entrée de la table de correspondances en écriture (figure 1). Cette structure plus complexe se prête également à des fonctionnalités moins standard. La fonction « MEM OP », par exemple, permet aux microcontrôleurs de sauvegarder en mémoire la configuration des E/S à usage général (GPIO) courante, ou de charger une autre configuration GPIO depuis la mémoire. MEM OP peut également réinitialiser la configuration GPIO selon les paramètres définis lors de la compilation.

En option, le microcontrôleur peut également être paramétré pour charger les réglages sauvegardés lors de l'allumage. Si cette option est activée, le microcontrôleur essaie de charger les paramètres en configuration 0. Si la configuration échoue lors de la validation de la somme de contrôle, alors le microcontrôleur reviendra aux constantes définies au moment de la compilation. Cette fonctionnalité peut être désactivée dans le logiciel si elle s'avère inutile. Contrairement aux ASIC disponibles sur le marché, le microcontrôleur peut donc être configuré avec des fonctionnalités non standard, spécifiques à l'application utilisée. Cette application a été développée pour la famille

1 TABLE DE CORRESPONDANCES D'UN MODULE D'EXTENSION D'E/S AVANCÉ BÂTI SUR UN MICROCONTRÔLEUR 8 BITS

Les entrées dans la table de correspondances peuvent être facilement réorganisées pour différents usages.

7	0	
0x00	ERROR	Get Error Code
0x01	IOCx	Interrupt-On-Change (IOC) Flags
0x02	PORTx	Current I/O Levels
0x03	TRISx	Tri-State Control Register
0x04	LATx	Output Latch Register
0x05	IOCxP	Enable IOC on Rising Edges
0x06	IOCxN	Enable IOC on Falling Edges
0x08	WPUx	Weak Pull-Up Control
0x09	INLVLx	Input Level Threshold
0x0A	ODCONx	Open-Drain Control
0x0B	SLRCONx	Slew Rate Control
0xA0	MEM OP	Memory Operation to Execute
	UNLOCK 1	
	UNLOCK 2	
		Unlock Sequence
0xB0	ADR Update	Updates I ² C Address
0xFF		

Legend

- Read Only
- Read and Write
- Write Only
- Write Only, Indirect Access
- Invalid
- Select Only

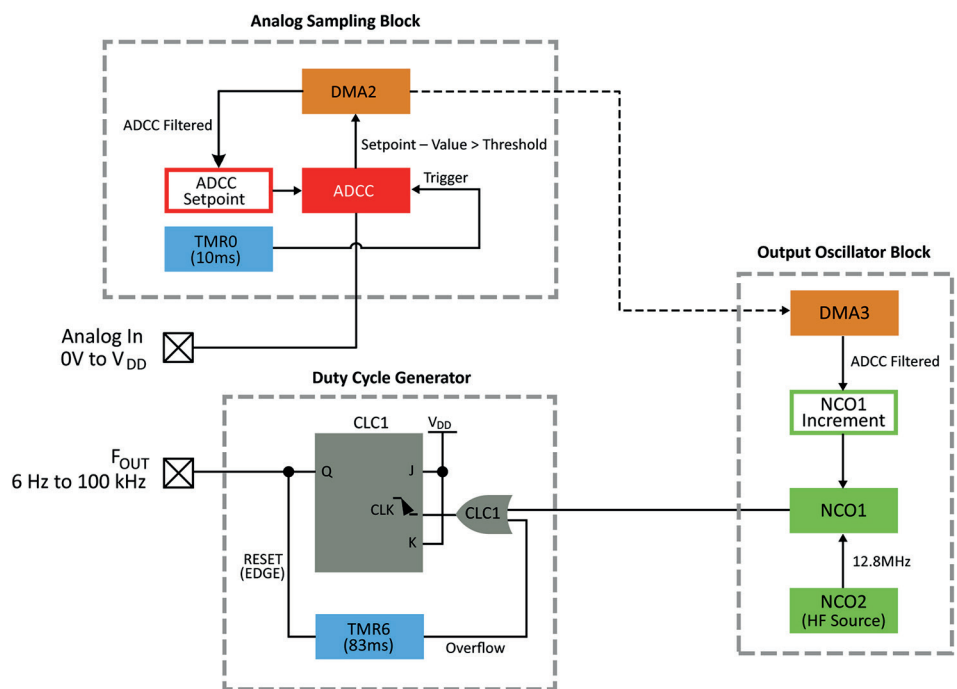
de microcontrôleurs PIC16F15244 à usage général. Le code source est disponible sur le site Microchip Discover (discover.microchip.com) et sur la plate-forme GitHub.

Cas du convertisseur tension-fréquence (V/F)

Le convertisseur tension-fréquence (V/F) illustre l'utilisation de périphériques matériels pour créer une fonction indépendante du cœur, qui devrait normalement être créée à l'aide d'un circuit intégré externe. L'un des principaux avantages de cette configuration est que les périphériques sont définis au niveau logiciel, ce qui facilite toute modification de l'exemple donné. Autre avantage: le convertisseur est mis en œuvre dans le microcontrôleur, ce qui réduit la liste de composants, ainsi que l'espace occupé par le système. L'un des problèmes de l'approche reposant sur un microcontrôleur est qu'elle ne s'avère pas aussi performante que la solution analogique correspondante, puisque la résolu-

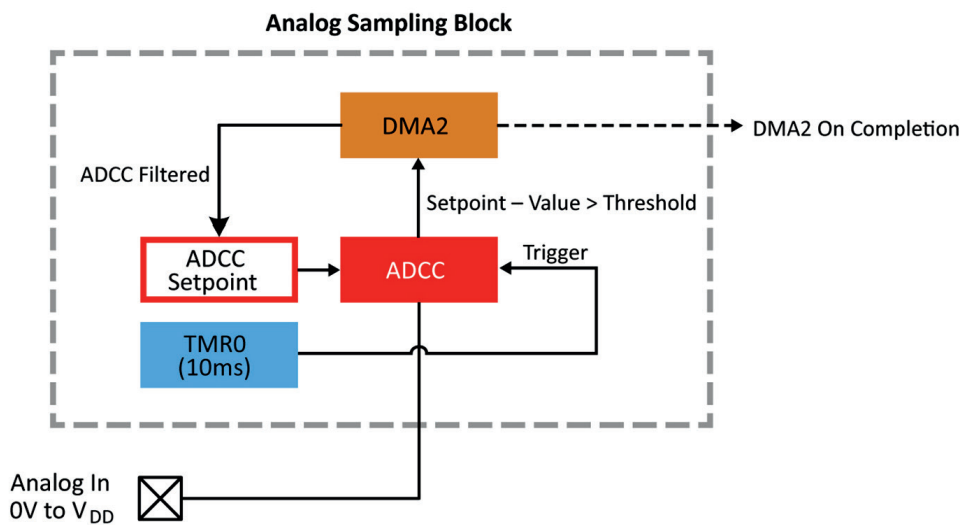
2 SCHÉMA ÉLECTRONIQUE DE LA SOLUTION DE CONVERSION TENSION-FRÉQUENCE

Pour mettre en œuvre cette solution sur un microcontrôleur, trois blocs de périphériques distincts sont nécessaires : un bloc d'échantillonnage analogique, un bloc de l'oscillateur de sortie et un générateur de rapports cycliques.



3 ARCHITECTURE DU BLOC D'ÉCHANTILLONNAGE ANALOGIQUE

Ce schéma détaille comment le bloc d'échantillonnage analogique est mis en œuvre dans une solution reposant sur un microcontrôleur



tion de la sortie est intrinsèquement limitée par le convertisseur analogique-numérique avec calcul (ADCC). Le convertisseur ADCC a une résolution nominale de 12 bits, mais il a été reconfiguré pour obtenir une résolution sur-échantillonnée de 14 bits. De même, l'oscillateur contrôlé numériquement (NCO) intégré, qui synthétise la fréquence de sortie, possède une résolution finie et peut avoir une gigue en sortie, en fonction de la valeur mesurée par le convertisseur N/A.

Pour mettre en œuvre cette solution, trois blocs de périphériques distincts sont nécessaires : un bloc d'échantillonnage analogique, un bloc de l'oscillateur de sortie et un générateur de rapports cycliques (figure 2).

Bloc d'échantillonnage analogique

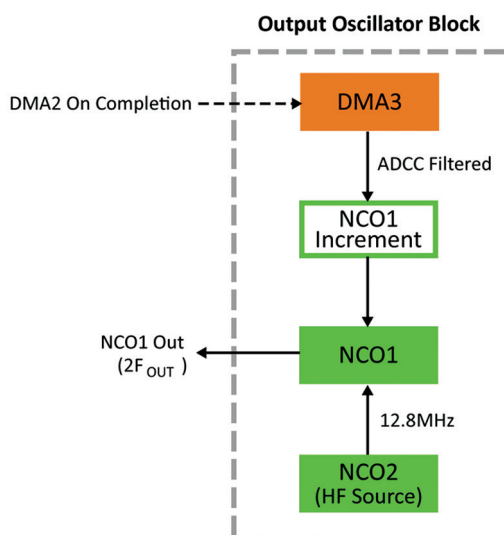
Le bloc d'échantillonnage analogique a pour rôle d'effectuer la conversion analogique-numérique (figure 3). Pour qu'une sortie de 100kHz soit possible avec les limites de fréquence du composant, le convertisseur ADCC a été configuré pour accumuler de nombreux échantillons puis réaliser une moyenne pour obtenir un résultat sur-échantillonné de 14 bits.

L'inconvénient de ce sur-échantillonnage est le bruit statistique supplémentaire ajouté au résultat. Le bruit est compensé par la moyenne réalisée à partir du sur-échantillonnage à laquelle on ajoute l'hystérésis. Pour

mettre en œuvre l'hystérésis, on utilise la fonction d'interruption liée à un seuil du convertisseur ADCC. (Pour plus de simplicité, seules seront détaillées les spécificités expliquant comment cette fonction d'interruption liée à un seuil est utilisée dans notre exemple.) Quand le convertisseur ADCC finit le calcul de la moyenne à partir du sur-échantillonnage, la valeur du résultat est comparée à la valeur consignée dans le registre du périphérique. Si la différence entre les deux valeurs est supérieure ou inférieure aux limites fixées,

4 ARCHITECTURE DU BLOC DE L'OSCILLATEUR DE SORTIE

Le bloc de l'oscillateur de sortie de la solution sur microcontrôleur a pour rôle de générer un signal d'horloge à la fréquence de sortie désirée.



alors une interruption est déclenchée. Le CPU masque cette interruption et n'en est pas affecté; cependant, cette interruption déclenche une copie, dans la mémoire DMA, du résultat de la moyenne du sur-échantillonnage par rapport à la valeur consignée dans le registre du convertisseur ADCC, et elle entraîne une mise à jour le point médian de l'hystérésis. Si les limites ne sont pas dépassées, alors la copie en mémoire DMA ne se fait pas, ce qui ne déclenche pas la mise à jour de la DMA du bloc de l'oscillateur de sortie.

Bloc de l'oscillateur de sortie

Le bloc de l'oscillateur de sortie de cette solution (figure 4) a pour rôle de générer un signal d'horloge à la fréquence de sortie désirée. Le signal de sortie est connecté en interne au générateur de rapports cycliques, ce qui diminue de moitié la fréquence de la sortie mais crée une sortie de rapport cyclique de 50%. C'est pourquoi le bloc de l'oscillateur de sortie fonctionne à une fréquence deux fois supérieure à celle de la sortie.

Le cœur du bloc de l'oscillateur de sortie est un oscillateur contrôlé numériquement (NCO). Le périphérique NCO fonctionne en ajoutant une valeur incrémentale à un accumulateur sur le front ascendant d'une horloge d'entrée. Quand l'accumulateur dépasse sa capacité, la sortie du périphérique est dérivée.

Dans notre exemple, le NCO2 a été paramétré en interne afin de créer la fréquence d'horloge nécessaire en entrée, pour obtenir une sortie à 100kHz à partir d'une entrée 14 bits. Le résultat de 14 bits est utilisé parce que le résultat natif de 12 bits issu du convertisseur ADCC n'est pas assez grand pour générer une sortie 100 kHz sans source d'horloge externe (tableau A). Si la fréquence de sortie de NCO2 est modifiée, ou qu'une source alternative est utilisée, la fréquence de sortie sera redimensionnée pour une plage de sortie différente. Par exemple, si NCO2 est réduit à 1,28MHz, la sortie plafonne à 10kHz (tableau B).

Générateur de rapports cycliques

Le bloc du générateur de rapports cycliques de cette solution a pour rôle de créer une sortie de rapport

A SORTIES IDÉALES DU CONVERTISSEUR V/F 100 kHz AVEC WATCHDOG ÉTEINT

Résultat conv. A/N	Sortie NCO1 (doublée)	Fréquence de sortie
0x0000	0 Hz	0 Hz
0x0001	12,2 Hz	6,1 Hz
0x0100	3,1 kHz	1,6 kHz
0x1000	50 kHz	25 kHz
0x3FFF	200 kHz	100 kHz

B SORTIES IDÉALES DU CONVERTISSEUR V/F 10 kHz AVEC WTECHDOG ÉTEINT

Résultat conv. A/N	Fréquence de sortie NCO1 (doublée)	Fréquence de sortie
0x0000	0 Hz	0 Hz
0x0001	1,2 Hz	0,6 Hz
0x0100	312,5 Hz	156,3 Hz
0x1000	5 kHz	2,5 kHz
0x3FFF	20 kHz	10 kHz

cyclique de 50% (Figure 5). Cette fonctionnalité est optionnelle. La sortie du NCO peut être utilisée directement, bien que le rapport cyclique varie dans des proportions bien plus grandes.

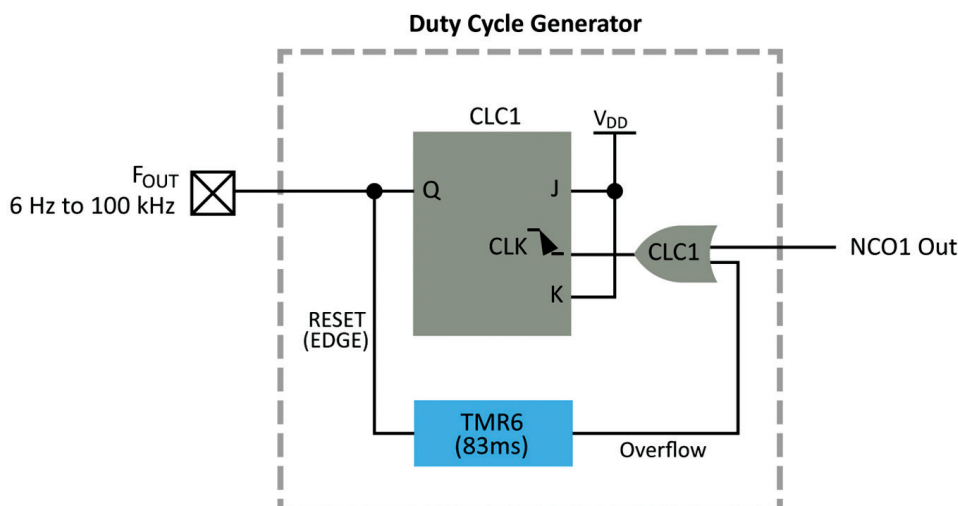
Ce générateur est mis en œuvre à l'aide de l'une des cellules CLC (Configurable Logic Cells). Les cellules CLC sont de petits blocs de logique configurable, semblables à une cellule d'un FPGA (Field-Programmable Gate Array, soit Réseau de portes programmables in situ). Les cellules CLC peuvent être utilisées comme portes logiques discrètes,

telles que AND-OR ou OR-XOR. Elles peuvent également être configurées comme des verrous ou circuits à bascules. Dans cette solution, une cellule CLC a été mise en œuvre en tant que bascule J-K avec réinitialisation. J et K sont maintenus à l'état logique HAUT. Le bloc de l'oscillateur de sortie est utilisé comme horloge pour la bascule. Chaque impulsion de l'horloge d'entrée fait commuter la sortie, ce qui génère un rapport cyclique de 50%. Remarque : La gigue de fréquence à partir du bloc de l'oscillateur de sortie affecte le rapport cyclique.

Le Timer 6 est utilisé comme temporisateur de watchdog instable. Si la sortie ne produit pas de front (ascendant ou descendant), alors le temporisateur dépassera sa capacité et générera une impulsion d'horloge envoyée à la cellule CLC. Le côté bas de la plage de fréquence de sortie est contrôlé de cette façon. Plutôt que d'atteindre le courant DC, la sortie commute à la moitié de la fréquence du temporisateur, pour obtenir une sortie de 6 Hz. Pour mettre en œuvre l'exemple, la famille de microcontrôleurs PIC18-Q43 a été sélectionnée en raison du grand nombre de périphériques qu'elle possède. Le code source de ce projet est disponible sur le site Microchip Discover (discover.microchip.com) et GitHub.

5 ARCHITECTURE DU GÉNÉRATEUR DE RAPPORTS CYCLIQUES

Le bloc du générateur de rapports cycliques de la solution sur microcontrôleur a pour rôle de créer une sortie de rapport cyclique de 50%.

**Conclusion**

Tandis que les microcontrôleurs et microprocesseurs hautes performances sont voués à rester sur le marché, les microcontrôleurs 8 et 16 bits représentent un outil inestimable pour réaliser de petites tâches spécialisées. Les tâches en question ne sont pas nécessairement complexes, mais elles peuvent s'avérer chronophages ou supposer des contraintes de temps. En se déchargeant de ces tâches, les composants 32 bits peuvent bénéficier d'une mise en œuvre plus simple, qui améliore la fiabilité, réduit l'utilisation de la mémoire, et diminue la consommation d'énergie. ■