

FAQ semaine 7

1) *Quelle est la différence entre TA0CTL et TACTL. Est-ce que ce sont deux registres de contrôle différent du timer A ?*

Réponse : TACTL est le nom donné au registre de contrôle du Timer A sur les microcontrôleurs n'ayant qu'un timer. Sur les modèles qui en ont plusieurs, il est appelé TA0CTL sur le premier timer, TA1CTL, TA2CTL sur les suivants.

Idem pour les registres TA0CCTL0, TA0CCTL1, ..., TA1CCTL0, TA1CCTL1, etc.

2) *Pourquoi utilise-t-on les interruptions ? Je ne comprends pas trop l'intérêt...*

Réponse : La motivation des interruptions est d'éviter le gaspillage de ressource du processeur et la possible perte de temps liés à la scrutation d'événements. Avec les interruptions, ce sont les événements qui se signalent au processeur.

3) *Pourquoi utiliser plutôt pull-up que pull-down pour les entrées ?*

Réponse : On utilise un pull-up lorsque l'état 0 est assuré par le matériel connecté à l'entrée (par exemple un poussoir connecté contre la masse), mais que l'état 1 n'est pas donné par le matériel (poussoir en l'air). Comme c'est ce schéma qui est utilisé pour les poussoirs des cartes MSP430, on doit l'utiliser.

On utilise un pull-down lorsque l'état 1 est assuré par le matériel connecté à l'entrée (par exemple un poussoir connecté vers le +), mais que l'état 0 n'est pas donné par le matériel (poussoir en l'air).

4) *Pour le TP sur les Timers, j'ai du mal à comprendre la syntaxe pour l'appelle des registres de timer et comparaison vu que il est pas la même dans les exemples présentés dans le cours.*

Réponse : En effet, le cours avait été écrit pour un MSP430G, où Texas Instrument autorisait une syntaxe simplifiée, correspondant aux processeurs n'ayant qu'un Timer A. Les registres commençant par TA... sans chiffre doivent maintenant être notés TA0...

Il faut bien différencier les registres de contrôle du Timer et les registres de contrôles des registres de comparaison :

- Registre de contrôle du timer : TA0CTL (pour le timer 0, et aussi TA1CTL pour le timer 1).
- Registre de contrôle des registres de comparaison : TA0CCTL0, TA0CCTL1, TA0CCTL2, TA0CCTL3, TA0CCTL4 (pour le timer 0, et aussi TA1CCTL0, TA1CCTL1, TA1CCTL2, pour le timer 1, qui n'a que trois registres de comparaison, tout comme le timer 2).

A noter : TA0CCTL est incorrect : on ne dit pas quel registre de comparaison est concerné. De même, TA0CTL0 est aussi incorrect, il n'y a qu'un registre de contrôle général pour un timer.

5) *Est-ce que TA0CTL = TASSEL_2 | ID_3 | MC_2 | TAIE;
est équivalent à TA0CTL = TASSEL_2 + ID_3 + MC_2 + TAIE;*

Réponse : Texas Instrument utilise la syntaxe : TA0CTL = TASSEL_2 + ID_3 + MC_2 + TAIE;
Ajouter des champs de bits indépendants donne un résultat correct.

Je préfère personnellement la syntaxe TA0CTL = TASSEL_2 | ID_3 | MC_2 | TAIE;

Il s'agit de champs de bit, c'est donc une opération logique qui est plus indiquée.

Mais le résultat est évidemment le même !

6) *Quelle est la différence entre le Timer0_A0 Interrupt Vector et le Timer0_A1 Interrupt Vector ?*

Réponse : Il s'agit de deux vecteurs d'interruptions correspondant au timer 0. Le Timer0_A0

Interrupt Vector correspond au seul événement du registre de comparaison 0. Le Timer0_A1 Interrupt Vector correspond à plusieurs événements : les registres de comparaison 1 à 4 et l'événement du timer lui-même (overflow). Le registre TA0IV permet de différencier ces événements dans la routine d'interruption correspondante.