

# TP N° 5

## Utilisation de l'interruption TMR0 via le PIC16F84A

### 5.1 Objectifs

- Comprendre le mécanisme de l'interruption TMR0.
- Comprendre l'utilisation du cycle d'horloge et le cycle d'instruction.
- Mise en œuvre de l'interruption TMR0 à base du microcontrôleur PIC16F84A.

### 5.2 Introduction

Dans le TP précédent nous avons déjà vu qu'il existe une interruption interne du Timer. Cette interruption utilise un registre de 8bits appelé TMR0 ou Timer0 qui s'incrémente à chaque instruction. Il est en quelque sorte, le fréquencemètre de notre Microcontrôleur. Plus la fréquence est élevée plus il compte vite, il a donc une relation avec le temps (Horloge). La figure 5.1 montre le registre TMR0 ainsi le mode de fonctionnement de l'interruption timer et l'ensemble des registres intervenants.

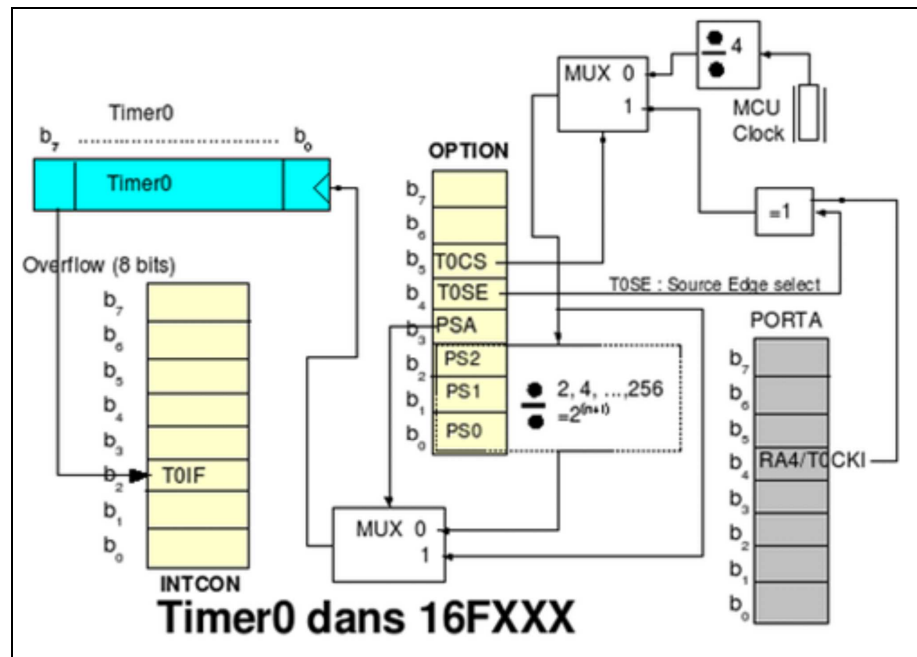


FIGURE 5.1: Les registres complémentaires pour le timer0

Les éléments remarquables dans la figure 5.1 :

- La fréquence du quartz est divisée par 4 avant utilisation (en haut à droite)
- C'est le registre OPTION qui permet une configuration du fonctionnement du TIMER0
- Lors d'un débordement (overflow) du timer 0, un bit nommé "T0IF" du registre INTCON est positionné à 1. Ce bit n'est jamais remis à 0 par le matériel car c'est un drapeau (flag).
- Le bit "T0CS" du registre OPTION permet, à l'aide du multiplexeur en haut à droite de choisir l'entrée du timer 0. On voit qu'il peut s'agir du quartz interne du PIC ou du bit "T0CKI" du PORTA.
- Le ou exclusif (=1) permet une éventuelle inversion
- Les trois bits "PS2", "PS1" et "PS0" permettent un choix de division entre 2 et 256
- La fameuse division ci-dessus ne sera prise en compte que si le multiplexeur du bas est positionné correctement par le bit "PSA" du registre OPTION.
- Cette figure vous donne aussi la position de tous les bits. Par exemple, le fameux bit "T0CKI" du PORTA est le bit RA4/T0CKI du PORTA.

Le tableau suivant présente la structure du registre **OPTION** (OPTION\_REG)

	<b>Nom</b>	<b>Description</b>
bit 7	NOT_RBPU	<p><b>"Port B Pull-up Enable"</b></p> <ul style="list-style-type: none"> <li>- Ce bit doit être mis à 0 pour <b>activer les résistances de pull-up du port B.</b></li> <li>- Ce bit doit être mis à 1 pour désactiver les résistances de pull-up du port B.</li> </ul>
bit 6	INTEDG	<p><b>"Interruptedge select"</b></p> <ul style="list-style-type: none"> <li>- Ce bit doit être mis à 0 pour que l'interruption de la broche RB0/INT soit active sur un front descendant.</li> <li>- Ce bit doit être mis à 1 pour que l'interruption de la broche RB0/INT soit active sur un front montant.</li> </ul>
bit 5	T0CS	<p><b>"TMR0 Clock Source Select"</b></p> <ul style="list-style-type: none"> <li>- Ce bit doit être mis à 0 pour que l'horloge du module TMR0 soit l'horloge interne (un quart de la fréquence du signal OSC1/CLKIN).</li> <li>- Ce bit doit être mis à 1 pour que l'horloge du module TMR0 soit le signal de la broche RA4/T0CKI.</li> </ul>
bit 4	T0SE	<p><b>"TMR0 Source edge select"</b></p> <p>Dans le cas où T0CS = 1, le signal d'horloge de la broche RA4/T0CKI est actif :</p> <ul style="list-style-type: none"> <li>- Sur front montant quand T0SE = 0.</li> <li>- Sur front descendant quand T0SE = 1.</li> </ul>
bit 3	PSA	<p><b>"Prescalerassignment"</b></p> <p>Le prédiviseur est attribué :</p> <ul style="list-style-type: none"> <li>- Au module TMR0 quand PSA = 0</li> </ul>

		-Au Watchdog quand PSA = 1
bits 2, 1, 0	PS2, PS1, PS0	<b>"Prescaler rate select"</b>

TABLE 5.1: Description de registre OPTION\_REG

### 5.2.1 Paramètres du Prescaler

Pour configurer le taux de pré-division du Timer TMR0, vous pouvez définir les champs PSA, PS2, PS1 et PS0 dans le registre OPTION comme le montre le tableau 5.2.

PSA	PS2, PS1, PS0	Taux de pré-division du module TMR0
0	000	2
0	001	4
0	010	8
0	011	16
0	100	32
0	101	64
0	110	128
0	111	256

TABLE 5.2: Paramètres du Prescaler

## 5.2.2 Principe de fonctionnement

Pour mieux comprendre la configuration du TMR0 et son fonctionnement, nous allons supposer que notre MicroContrôleur fonctionne à une fréquence de 8 Mhz.

Il faut savoir que le microcontrôleur a besoin de 4 cycles d'horloge, pour faire une instruction.

Donc le temps d'exécution d'une instruction =  $(1/8 \text{ Mhz}) * 4 = 0.5 \mu\text{s}$ .

Alors, notre compteur va s'incrémenter toutes les  $0.5 \mu\text{s}$ .

Cela va tellement vite, que dans de nombreux cas, il va encore falloir ralentir notre Timer. Pour cela on utilise ce qu'on appelle un Prescaler, qui va encore diviser notre fréquence. Ce Prescaler est réglable par la configuration du registre OPTION (PSA, PS2, PS1, PS0).

Si on choisit de diviser notre fréquence par 256 (PSA=0, PS2=1, PS1=1, PS0=1), c'est-à-dire qu'en fonction du temps nous allons multiplier la période par 256.

$$0.5 \mu\text{s} * 256 = 128 \mu\text{s}.$$

Cela veut dire, que si nous avons un oscillateur à 8 Mhz, avec un prescaler réglé sur 256 nous allons avoir une instruction qui s'exécute toutes les  $128 \mu\text{s}$ , donc notre compteur(TMR0) va s'incrémenter toutes les  $128 \mu\text{s}$ .

Sachant que notre compteur TMR0 est un registre de 8bits,il pourra compter 256 instructions avant de mettre le Flag T0IF à 1. Ce qui implique une interruption chaque  $128 \mu\text{s} * 256 = 32.768 \text{ m s}$ .

Si à chaque interruption du Timer on incrémente un compteur, il faudra 31 incrémentation de ce compteur pour obtenir approximativement une seconde (1000 ms) ( $1000/32.768 = 30.51 \approx 31$  fois).

### 5.3 Travail à réaliser

Dans ce qui suit, nous allons reprendre l'application 3 du TP précédant pour réaliser un chronomètre simplifié en utilisant trois afficheurs 7-segments BCD et un bouton connecté à la broche RB0. Le temps doit s'afficher sur les trois afficheurs 7-segments sous le format : « m – s s » (où m représente les minutes et s représente les secondes).

1. Garder le même mode de fonctionnement et remplacer la fonction delay\_ms par l'utilisation du timer0.
2. Pour améliorer votre chronomètre modifier le schéma en utilisant le composant 7SEG-MPX6-CA (voir figure 5.2) et le composant 4511 (voir figure 5.3).

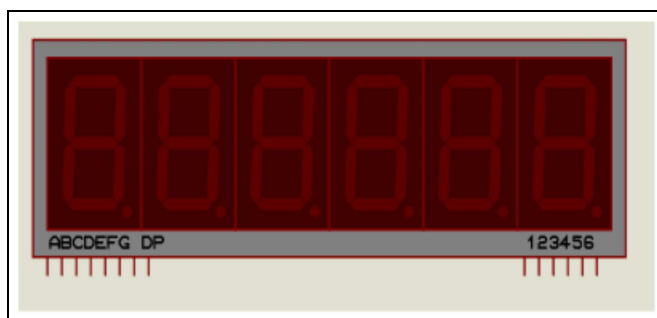


FIGURE 5.2: Afficheur 7 segments de type "7SEG-MPX6-CA"

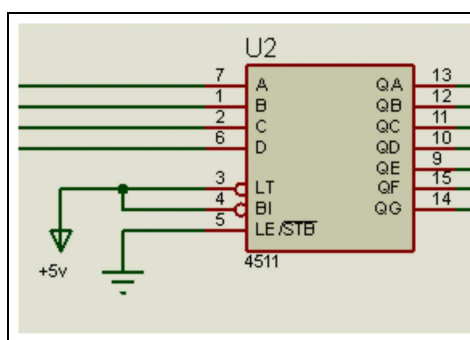


FIGURE 5.3: Composant 4511