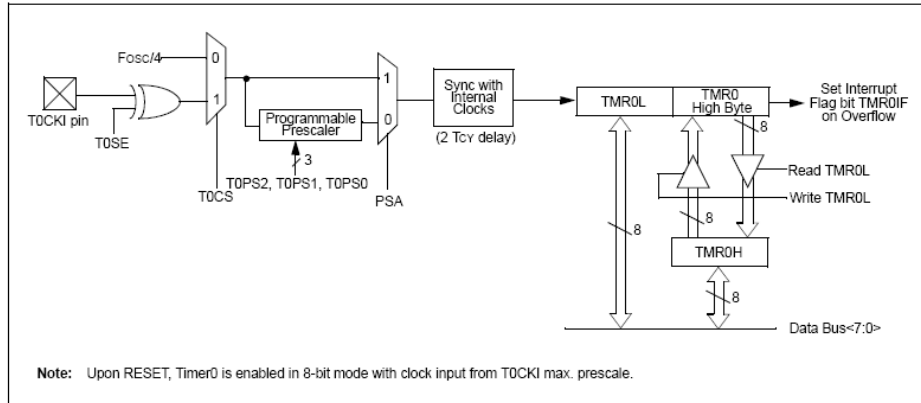


### Objectif :

- Faire clignoter une ou plusieurs led en utilisant le Timer 0 du microcontrôleur.

### Principe de fonctionnement du Timer 0 :

FIGURE 10-2: TIMER0 BLOCK DIAGRAM IN 16-BIT MODE



REGISTER 10-1: T0CON: TIMER0 CONTROL REGISTER

	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
	TMR0ON	T08BIT	T0CS	T0SE	PSA	T0PS2	T0PS1	T0PS0
bit 7								bit 0
bit 7	<b>TMR0ON:</b> Timer0 On/Off Control bit 1 = Enables Timer0 0 = Stops Timer0							
bit 6	<b>T08BIT:</b> Timer0 8-bit/16-bit Control bit 1 = Timer0 is configured as an 8-bit timer/counter 0 = Timer0 is configured as a 16-bit timer/counter							
bit 5	<b>T0CS:</b> Timer0 Clock Source Select bit 1 = Transition on T0CKI pin 0 = Internal instruction cycle clock (CLKO)							
bit 4	<b>T0SE:</b> Timer0 Source Edge Select bit 1 = Increment on high-to-low transition on T0CKI pin 0 = Increment on low-to-high transition on T0CKI pin							
bit 3	<b>PSA:</b> Timer0 Prescaler Assignment bit 1 = Timer0 prescaler is NOT assigned. Timer0 clock input bypasses prescaler. 0 = Timer0 prescaler is assigned. Timer0 clock input comes from prescaler output.							
bit 2-0	<b>T0PS2:T0PS0:</b> Timer0 Prescaler Select bits 111 = 1:256 prescale value 110 = 1:128 prescale value 101 = 1:64 prescale value 100 = 1:32 prescale value 011 = 1:16 prescale value 010 = 1:8 prescale value 001 = 1:4 prescale value 000 = 1:2 prescale value							
Legend: R = Readable bit      W = Writable bit      U = Unimplemented bit, read as '0' - n = Value at POR      '1' = Bit is set      '0' = Bit is cleared      x = Bit is unknown								

Le Timer est un module du microcontrôleur qui se comporte comme un compteur et dont la fréquence de comptage est un multiple de la fréquence horloge du microcontrôleur (voir fréquence du Quartz). Il compte sans jamais s'arrêter. Lorsque il atteint sa capacité maximale (en mode 8 bits ce sera 255) il passe à la valeur 0 puis continue de compter. Au moment où il passe par la valeur 0 il met à 1 le bit TMR0IF du registre INTCON pour indiquer qu'il y eu dépassement de capacité.

On a la possibilité de précharger le Timer grâce aux registres TMR0H et TMR0L

- Q1 - Quelle est la capacité (valeur maximale) du Timer 0 si on le configure en Timer 16 bits?
- Q2 - Quel est le bit du registre T0CON qui permet de configurer le Timer en mode 16 bits et quel doit être sa valeur?
- Q3 - Quel est le bit du registre T0CON qui permet de sélectionner l'horloge interne "Fosc / 4" et quelle doit être sa valeur?
- Q4 - Quel est le bit du registre T0CON qui permet de sélectionner le prédiviseur et quel doit être sa valeur?
- Q5 - Sachant que Fosc = 16 MHz, Combien de cycle doit compter le Timer0 pour obtenir une période de 1s.
- Q6 - Faut-il utiliser le prédiviseur et si oui quelle valeur de prédivision suggérez-vous pour obtenir une période de 1s?

Q7 - En utilisant une prédiviseur de 128, quel sont les valeurs à précharger dans TMR0H et TMR0L?

Q8 - Ouvrir le projet MPLAB : "Led clignotante Timer", puis compiler, télécharger et exécuter le programme. Que constatez – vous?

**Vérifier les "Build Options" :** Voir TP0 et TP1

Q9 - En tenant compte des questions Q5 à Q7, modifier le fichier "Tempo.c" pour obtenir un clignotement toutes les 1 s. Vous devez modifier les bits TOPS2, TOPS1, TOPS0 et les registres TMR0H et TMR0L.



Q10 - Donner pour les 4 algorithmes suivants le nom du fichier écrit en C ainsi que le nom de la fonction.

