

3IRC – Mise en œuvre d'un système à microprocesseur

Etude des périphériques dans un microcontrôleur 8051F020 – 2

Timers -- Version 2022



Introduction aux Timers

Etude conduite autour du Timer2 du 8051F020

Les timers: quel usage?

Les Timers (temporisateurs) sont des périphériques du microcontrôleur. Ils sont mis en œuvre pour toute activité liée à la gestion du temps. Ils peuvent interagir avec le monde extérieur (au travers de broches PIO) et peuvent produire des interruptions.

Les Timers sont indispensables pour les besoins suivants:

- Gestion du temps Base de temps
- Mesure du temps écoulé
- Comptage d'évènements
- Génération de signaux



Timer pas à pas – Le registre de comptage

Le cœur du timer: 1 registre de comptage.

- Ce registre peut être un registre 8 bits ou être constitué de 2 registres 8 bits concaténés pour produire un registre 16 bits.
- Comme tout registre on peut lire ou écrire son contenu (dans le cas d'un registre 16 bits, l'accès pourra se faire à partir de 2 accès 8 bits successifs)

Exemple du Timer2: 2 registres TL2 (poids faible) et TH2 (Poids fort)

$$TL2 = 0x12;$$

TH2 = 0x24;

TL2	TH2
-----	-----

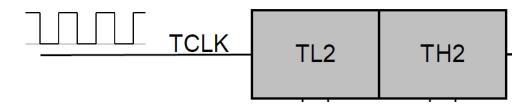
A l'issue de ces 2 lignes de codes, le contenu de ce registre 16 bits (constitué par la concaténation de TH2 et TL2) sera égale à 0x2412

• Mais le contenu de ce registre de comptage peut évoluer au cours du temps sans que l'on soit obligé d'écrire une nouvelle valeur.

Timer pas à pas – L'horloge Timer

L'horloge Timer.

- L'évolution du contenu des registres de comptage est commandé par un signal d'horloge connecté sur ces registres de comptage.
- C'est-à-dire que le changement de la valeur des registres de comptage sera rythmé par ce signal d'horloge
- Ce changement de valeur est soit une incrémentation, soit une décrémentation selon les possibilités du Timer (en 8051, les Timers ne fonctionnent qu'en incrémentation).



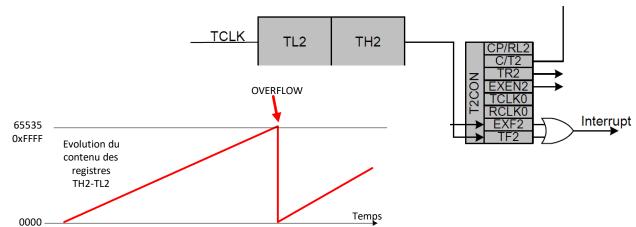
- Ainsi à chaque front négatif d'horloge TCLK la valeur du registre de comptage s'incrémente de 1 (cas des Timers du 8051)
- Dans le cas d'un registre 16 bits, il est donc possible de compter de 0 à 2^{16} -1, soit 0 à 65535. Arrivé à 65535, le prochain incrément fait repasser le contenu du registre de comptage à zéro.



Timer pas à pas – L'évènement Overflow

L'évènement « Overflow » (Dépassement).

- Le passage du registre de comptage de sa valeur maximale (65535 pour un registre 16 bits) à zéro est appelé « dépassement » (« Overflow ») et il constitue un évènement particulier.
- Cet « Overflow » peut provoquer une interruption.La transmission de cet évènement au mécanisme de gestion des interruptions n'est pas direct et passe par un registre intermédiaire: le registre de contrôle du Timer (T2CON dans le cas du Timer2 du 8051)



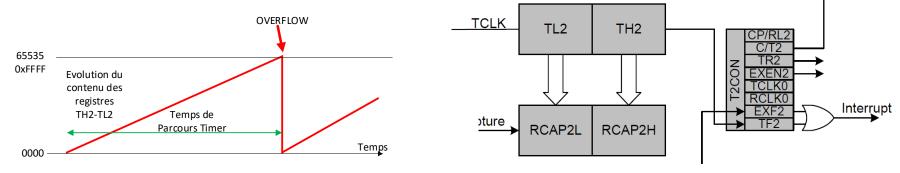
- L'évènement d'overflow est en quelque sorté mémorisé par un bit dans ce **registre de contrôle**. Dans le cas du 8051 et du Timer 2, l'évènement d'overflow met à un 1, le bit TF2 du registre T2CON
- Le rôle du registre de contrôle est de configurer, spécifier les modes de fonctionnement du Timer et de donner des renseignements sur l'état du timer (les évènements)



Timer pas à pas – Overflow et Interruption

Une interruption à chaque overflow

- Le Timer (le Timer2 dans cet exemple) peut donc produire une demande d'interruption, liée à un évènement d'overflow (bit TF2 de T2CON). Cette interruption sera traitée si l'interruption de ce Timer a été autorisée.
- Rappelons que « traiter » une interruption signifie que l'on va pouvoir faire exécuter le programme d'interruption lié à ce périphérique.
- Remarque: la mise en œuvre des interruptions est traitée dans un chapitre précédent du cours
- Grâce à ce mécanisme, on peut donc produire des interruptions périodiques. On a donc créé une base de temps de référence, car à chaque interruption, on connaîtra le temps écoulé depuis l'interruption précédente.

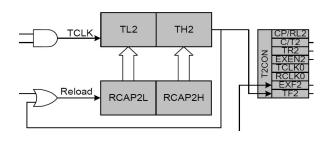


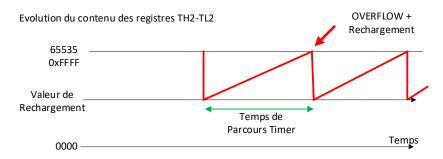
- Ce temps écoulé entre 2 interruptions Overflow Timer est lié à deux facteurs:
 - La fréquence de l'horloge CLK Timer
 - La résolution en nombre de bits du Timer
 - Dans le cas du Timer2 en prenant une fréquence d'horloge Timer de 1MHz, le temps écoulé entre 2 interruptions sera de 1μs X 65536 = 65,5ms
- Cet durée figée, n'est pas forcément idéale selon les applications, et la valeur obtenue dans l'exemple précédent ne simplifie pas les calculs de temps.

Timer pas à pas – Mode Auto-rechargement

L'autorechargement

- Pour modifier le temps de parcours du timer (le temps mis pour compter de sa valeur minimale à sa valeur maximale, c'est-à-dire de 0 à 65535 pour le Timer2), on va chercher à modifier la valeur minimale de comptage du registre de comptage.
- Au lieu de compter en continu de 0 à 65535, on va mettre en place un dispositif qui va permettre de compter d'une valeur programmable « Reload Value » (valeur minimale) jusqu'à 65535 (valeur maximale). Ainsi lorsque que le timer arrivera à sa valeur maximale, au lieu de répartir à zéro, il partira de la valeur de rechargement.
- Cette valeur de rechargement (Relaod Value) sera contenue dans les registres RCAP2H/RCAP2L dans le cas du Timer2 du 8051
- En changeant le contenu de ces 2 registres RCAP2H/RCAP2L, on modifie donc le temps de parcours du timer, et donc pas voie de conséquence, on change la récurrence de l'interruption « Overflow du Timer





- Ce temps écoulé entre 2 interruptions Overflow Timer est lié à deux facteurs:
 - La fréquence de l'horloge CLK Timer
 - La résolution en nombre de bits du Timer (la valeur maximale)
 - La valeur de rechargement (la valeur minimale)
- Le temps de parcours du Timer est donc égal à:

(valeur max CP Timer – valeur min CP Timer)* Période CLK Timer



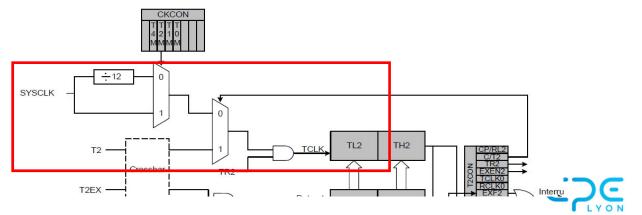
Timer pas à pas – Timer en Base de Temps

Le Timer en base de temps

- Compte tenu des éléments évoqués précédemment on dispose de toutes les informations pour mettre en œuvre une base de temps périodique, c'est-à-dire une interruption qui va se déclencher périodiquement.
 - Nous aurons la possibilité de régler cette périodicité
 - Le programme d'interruption permettra de gérer l'évolution temporelle des tâches diverses d'une application embarquée.
- Rappelons: Le temps de parcours du Timer est donc égal à (valeur max CP Timer
 – valeur min CP-Timer)* Période CLK Timer
- Action sur l'horloge CLK Timer?

L'horloge du timer (TCLK) peut être connectée à une source interne SYSCLK ou SYSCLK/12 ou à une source externe (branchée sur T2)

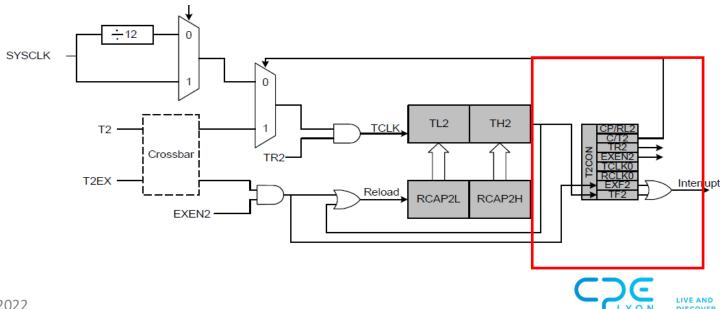
Pour des applications de type base de temps, on privilégiera la source d'horloge interne.



Timer pas à pas – Autres fonctionnalités

Fonctionnalités annexes du Timer

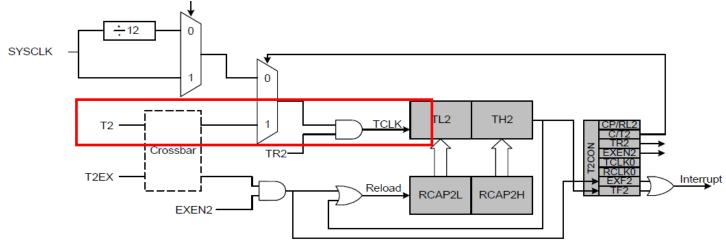
- Le registre de contrôle du Timer permet d'apporter d'autres possibilités au Timer:
- L'arrêt ou la mise en route du Timer (Bit TR2 pour le Timer2)
- Le choix d'horloge du Timer: interne ou externe
- Le choix du mode auto-rechargement ou capture



Timer pas à pas – Choix d'une Horloge Externe

Choix d'une horloge externe pour produire l'horloge Timer

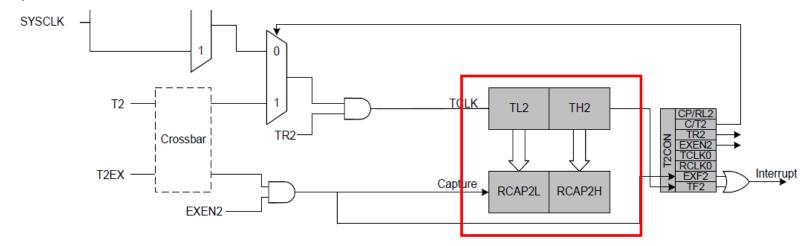
- En général, la source d'horloge externe est utilisée pour du comptage d'évènements. Chaque front actif d'horloge est alors considéré comme 1 évènement (dans la documentation, ce mode est appelé « Mode Compteur »)
- Dans le cas du Timer2, c'est la broche PIO T2 qui peut servir d'entrée d'horloge
- Associé, à un mode auto-rechargement, il est alors possible de déclencher des interruptions à chaque fois que l'on aura compté N



Timer pas à pas – Le mode « Capture »

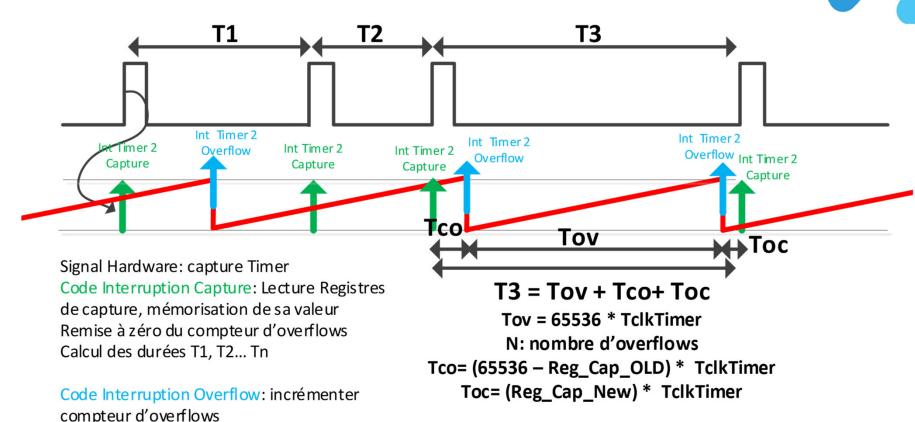
Le mode « Capture »

- Il est incompatible avec le mode autorechargement, car il exploite les registres RCAP2H-RCAP2L (Reload Capture) différemment
- Une capture consiste à « capturer » le contenu du registre de stockage à un instant donné et à stocker cette valeur dans les registres de Capture, et ce, jusqu'à la prochaine capture
- Dans le Timer2, cet évènement « Capture » est déclenché par un signal externe connecté sur l'entrée T2EX
- Cette capture est aussi susceptible de provoquer une interruption en positionnant le bit EXF2 à 1



Timer pas à pas – Utilisation du mode Capture

- •Timer configuré en mode Capture et comptage « Timer » (CLK interne)
- •Une seule interruption Timer, mais 2 sources possibles (Capture et Overflow)



Généralisation:

Tn = ((N*65536) + New_RCAP – Old_RCAP) * TclkTimer Avec N: nombre d'overflows depuis la Capture précédente New_RCAP : valeur des registres de capture pour la capture N Old_RCAP: valeur des registres de capture pour la capture N-1

Timer pas à pas – Bilan (simplifié)

Dans toute mise en œuvre de Timer, il convient de se poser les questions suivantes:

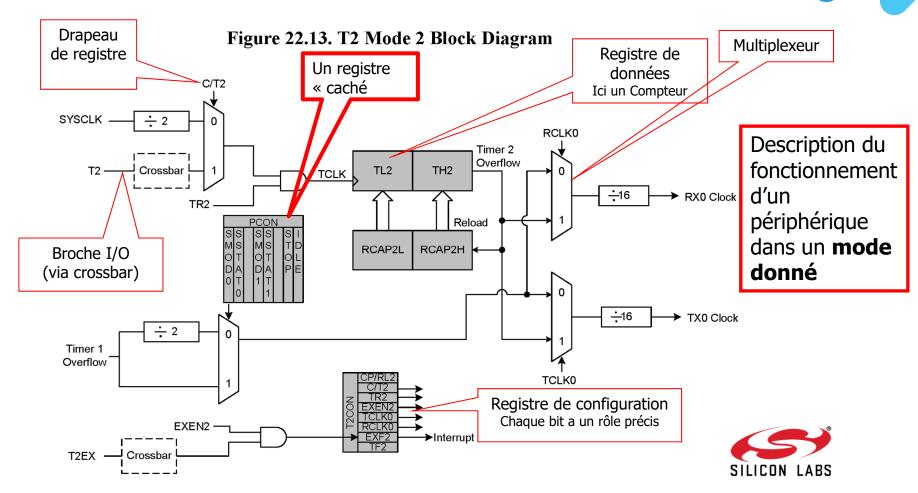
- Quelle horloge? Source d'horloge interne (Timer) ou externe (Compteur)
- Quel mode? mode Auto-rechargement, mode Capture, autre mode?
- Quel évènement? Quels sont les évènements produits par le timer (susceptibles de provoquer des interruptions) et qu'en faire?



Les Timers du 8051F020

Approfondissement sur la mise en œuvre du Timer2

Comprendre un « block-diagram » dans les documentations 8051F020



Descriptif d'un registre

Figure 22.5. TCON: Timer Control Register

R/W	R/W	Reset Value						
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1 _	Rit0	SFR Address:
57	77					(bit addressable)	0x88

Bit7: Time 1 Overflow Flag.

Set by hardware when Timer 1 overflows. This flag can be cleared by software but is automatically cleared when the CPU vectors to the Timer 1 interrupt service routine.

0: No Timer 1 overflow detected.

V. Timer 1 has overflowed.

Bit6: TX1: Timer 1 Run Control.

0: Timer 1 disabled.

1: Timer 1 enabled.

Bit5: TF0: Timer 0 Overflow Flag.

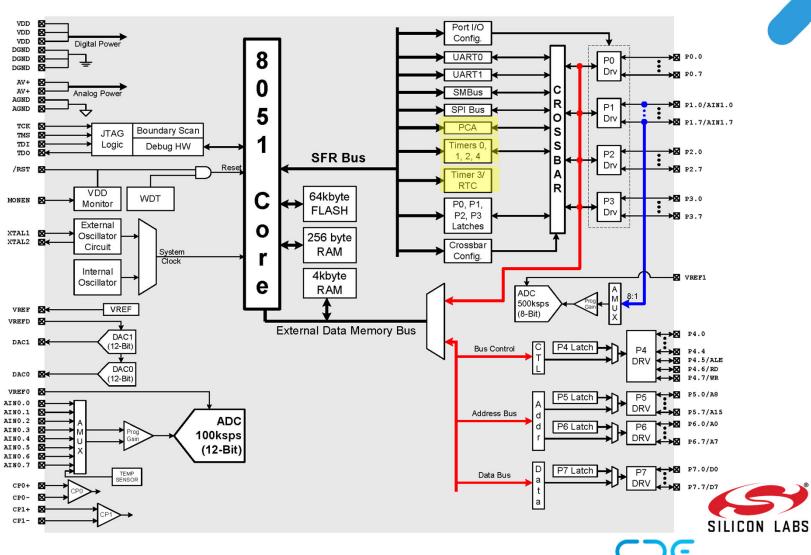
Set by hardware when Timer 0 overflows. This flag can be cleared by software but is automatically cleared when the CPU vectors to the Timer 0 interrupt service routine.

0: No Timer 0 overflow detected.

1: Timer 0 has overflowed.



Les Timers dans le 8051F020



Etude du Timer 2 dans le 8051F020

3 modes de fonctionnement:

- Mode 0 Compteur/Timer 16 bits avec capture
- Mode 1 Compteur/Timer 16 bits avec autorechargement
- Mode 2 générateur d'horloge pour UART 0

Plusieurs registres à manipuler:

1 registre de configuration et de contrôle T2CON

2 registres de comptage TH2 et TL2 pour former un compteur 16 bits

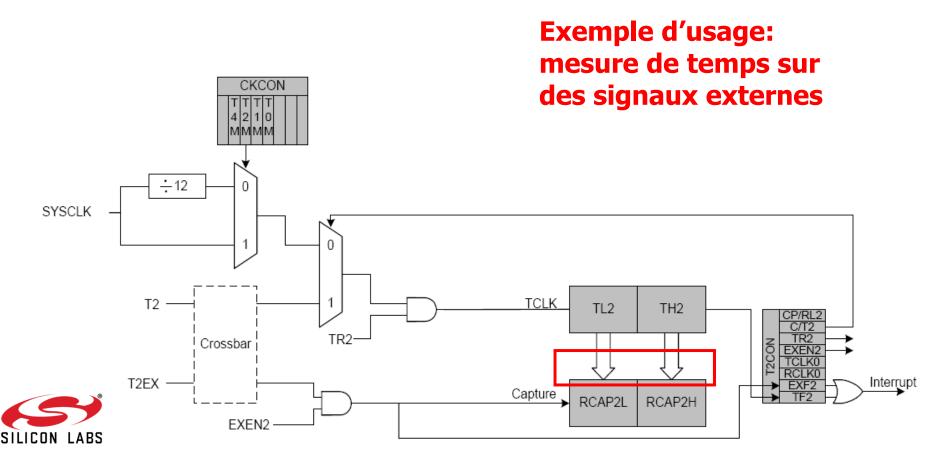
2 registres RCAP2L et RCAP2H

Quelques bits dans des registres divers (1 bit dans CKCON)



Timer 2 – Mode 0 - Capture

Compteur/Timer 16 bits avec capture: un front actif (front descendant) sur le signal externe T2EX provoque la recopie (la capture) du contenu des registres de comptage (TH2-TL2) dans les registres (RCAP2H-RCAP2L) sans interrompre le comptage.

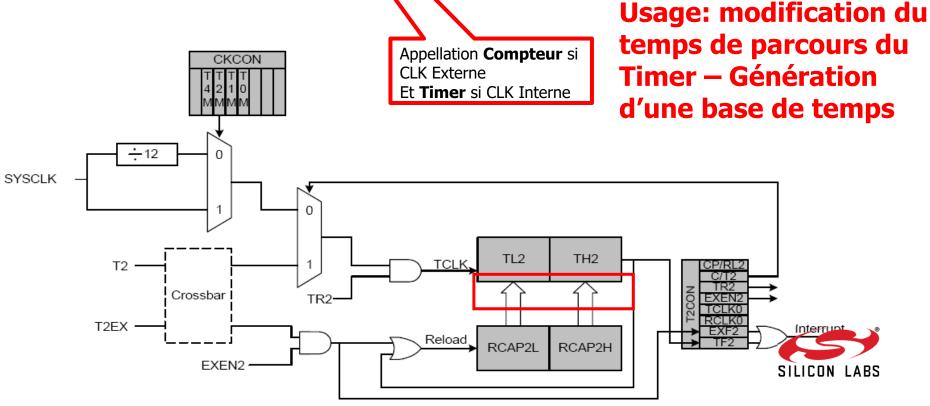


Timer 2 – Mode 1 - Autorechargement

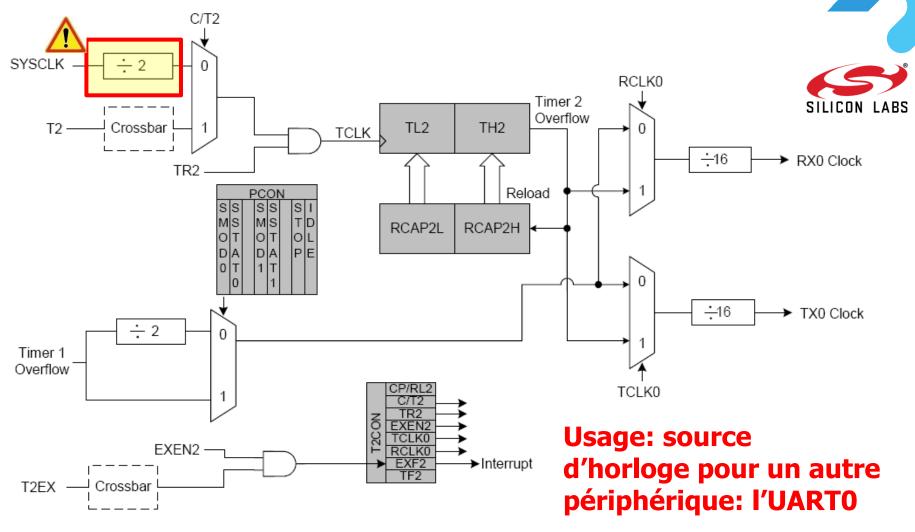
Compteur/Timer 16 bits avec auto rechargement: lorsque les registres de comptage (TH2-TL2) atteignent leur valeur maximale (0xFFFF), le contenu des registres RCAP2H-RCAP2L est recopié dans les registres de comptage.

Les registres du timer (TH2/TL2) évoluent en permanence de [RCAP2H-

RCAP2L] à 0xFFFF



Timer 2 – Mode 2 – Baud rate generator



Sélection du mode de fonctionnement du Timer 2

Les modes de fonctionnement du timer 2 sont définis à l'aide de 4 bits RCLKO, TCLKO, CP/RL2 et TR2 appartenant au registre de contrôle T2CON

	RCLK0	TCLK0	CP/RL2	TR2	Mode		
	0	0	1	1	16-bit Counter/Timer with Capture		
Ī	0	0	0	1	16-bit Counter/Timer with Auto-Reload		
	0	1	X	1	Baud Rate Generator for UART0		
	1	0	X	1	Baud Rate Generator for UART0		
	1	1	X	1	Baud Rate Generator for UART0		
	X	X	X	0	Off		

Registre T2CON_Bits 7-6-5

Registre T2CON: un registre de configuration et d'état

Figure 22.14. T2CON: Timer 2 Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
TF2	EXF2	RCLK0	TCLK0	EXEN2	TR2	C/T2	CP/RL2	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Rit()	SFR Address:
							bit addressable)	0xC8

Bit7: TF2: Timer 2 Overflow Flag.

Set by hardware when Timer 2 overflows. When the Timer 2 interrupt is enabled, setting this bit

Etat causes the CPU to vector to the Timer 2 interrupt service routine. This bit is not automatically cleared

by hardware and must be cleared by software. TF2 will not be set when RCLK0 and/or TCLK0 are

logic 1.

Etat

Bit6: EXF2: Timer 2 External Flag.

Set by hardware when either a capture or reload is caused by a high-to-low transition on the T2EX

input pin and EXEN2 is logic 1. When the Timer 2 interrupt is enabled, setting this bit causes the

CPU to vector to the Timer 2 Interrupt service routine. This bit is not automatically cleared by hard-

ware and must be cleared by software.

Bit5: RCLK0: Receive Clock Flag for UART0.

Selects which timer is used for the UART0 receive clock in modes 1 or 3.

Config 0: Timer 1 overflows used for receive clock.

1: Timer 2 overflows used for receive clock.



Registre T2CON Bits 4-3-2-1-0

Bit4: TCLK0: Transmit Clock Flag for UART0.

Selects which timer is used for the UART0 transmit clock in modes 1 or 3.

Config 0: Timer 1 overflows used for transmit clock.

1: Timer 2 overflows used for transmit clock.

Bit3: EXEN2: Timer 2 External Enable.

Enables high-to-low transitions on T2EX to trigger captures or reloads when Timer 2 is not operating

in Baud Rate Generator mode. Config

0: High-to-low transitions on T2EX ignored.

1: High-to-low transitions on T2EX cause a capture or reload.

Bit2: TR2: Timer 2 Run Control.

This bit enables/disables Timer 2.

Config 0: Timer 2 disabled.

1: Timer 2 enabled.

Bit1: C/T2: Counter/Timer Select.

0: Timer Function: Timer 2 incremented by clock defined by T2M (CKCON.5). Config

1: Counter Function: Timer 2 incremented by high-to-low transitions on external input pin (T2).

Bit0: CP/RL2: Capture/Reload Select.

This bit selects whether Timer 2 functions in capture or auto-reload mode. EXEN2 must be logic 1 for

high-to-low transitions on T2EX to be recognized and used to trigger captures or reloads. If RCLK0 **Config** or TCLK0 is set, this bit is ignored and Timer 2 will function in auto-reload mode.

0: Auto-reload on Timer 2 overflow or high-to-low transition at T2EX (EXEN2 = 1).

1: Capture on high-to-low transition at T2EX (EXEN2 = 1).



Registre CKCON – Un registre « multirôles »

Figure 22.1. CKCON: Clock Control Register

	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
	-	T4M	T2M	T1M	T0M	Reserved	Reserved	Reserved	00000000
ľ	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
									0x8E

Bit7: UNUSED. Read = 0b, Write = don't care.

Bit6: T4M: Timer 4 Clock Select.

This bit controls the division of the system clock supplied to Timer 4. This bit is ignored when the timer is in baud rate generator mode or counter mode (i.e. C/T4 = 1).

0: Timer 4 uses the system clock divided by 12.

1: Timer 4 uses the system clock.

Bit5: T2M: Timer 2 Clock Select.

This bit controls the division of the system clock supplied to Timer 2. This bit is ignored when the timer is in baud rate generator mode or counter mode (i.e. C/T2 = 1).

0: Timer 2 uses the system clock divided by 12.

1: Timer 2 uses the system clock.

Bit4: T1M: Timer 1 Clock Select.

This bit controls the division of the system clock supplied to Timer 1.

0: Timer 1 uses the system clock divided by 12.

1: Timer 1 uses the system clock.

Bit3: T0M: Timer 0 Clock Select.

This bit controls the division of the system clock supplied to Counter/Timer 0.

0: Counter/Timer uses the system clock divided by 12.

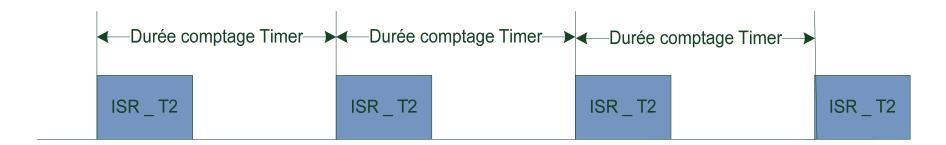
1: Counter/Timer uses the system clock.

Bits2-0: Reserved. Read = 000b, Must Write = 000.



Problématique du Timer 2 en base de temps

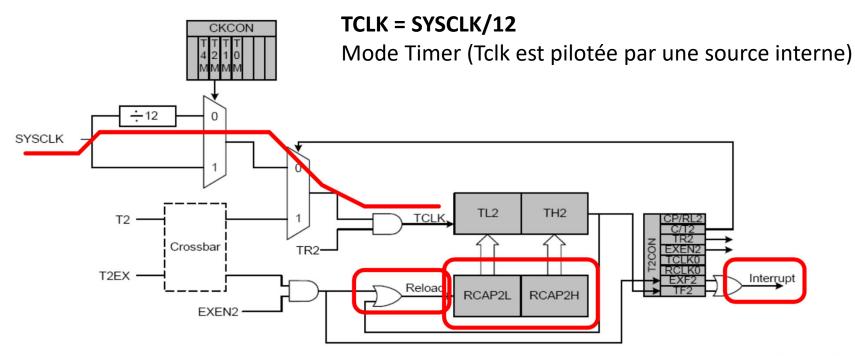
- Objectif: Génération d'une base de temps de période 10 milliseconde
- ➤ Déclencher une interruption ISR_T2 toutes les N secondes
- Contraintes: fréquence de l'interruption liée à la résolution du Timer et à sa fréquence de pilotage



Timer 2 en base de temps - Configuration

Choix du mode: Mode 1 - Timer en Auto-rechargement

Une fois configuré, le timer fait en continu des cycles de comptage de période voulue



Timer 2 en base de temps – Les registres impliqués

T2CON – Configuration principale du timer 2

CKCON – Configuration auxiliaire du Timer 2

RCAP2L et RCAP2H – Valeur de pré-chargement, elle permet de fixer la période de comptage du timer

TL2 et TH2 – les registres de comptage du Timer



Registre T2CON_Bits 5 6 7

Figure 22.14. T2CON: Timer 2 Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
TF2	EXF2	RCLK0	TCLK0	EXEN2	TR2	C/T2	CP/RL2	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Rit()	SFR Address:
							bit addressable)	0xC8

Bit7: TF2: Timer 2 Overflow Flag.

Set by hardware when Timer 2 overflows. When the Timer 2 interrupt is enabled, setting this bit

causes the CPU to vector to the Timer 2 interrupt service routine. This bit is not automatically cleared

by hardware and must be cleared by software. TF2 will not be set when RCLK0 and/or TCLK0 are

logic 1.

Bit6: EXF2: Timer 2 External Flag.

Set by hardware when either a capture or reload is caused by a high-to-low transition on the T2EX

input pin and EXEN2 is logic 1. When the Timer 2 interrupt is enabled, setting this bit causes the CPU to vector to the Timer 2 Interrust RCLK0 | TCLK0 | CP/RL2 | TR2 | Mode

ware and must be cleared by softwa

Bit5: RCLK0: Receive Clock Flag for UF

Selects which timer is used for the U

0: Timer 1 overflows used for receiv

1: Timer 2 overflows used for receiv

KCLKU	ICLKU	CP/KL2	1K2	Mode
0	0	1	1	16-bit Counter/Timer with Capture
0	0	0	1	16-bit Counter/Timer with Auto-Reload
0	1	X	1	Baud Rate Generator for UART0
1	0	X	1	Baud Rate Generator for UART0
1	1	X	1	Baud Rate Generator for UART0
X	X	X	0	Off

Registre T2CON Bits 0 1 2 3 4

Bit4: TCLK0: Transmit Clock Flag for UART0.

Selects which timer is used for the UART0 transmit clock in modes 1 or 3.

0: Timer 1 overflows used for transmit clock.

1: Timer 2 overflows used for transmit clock.

Bit3: EXEN2: Timer 2 External Enable.

Enables high-to-low transitions on T2EX to trigger captures or reloads when Timer 2 is not operating

TCLK0

0

CP/RL2

0

Χ

 \mathbf{X}

X

X

TR2

Mode

Off

in Baud Rate Generator mode.

0: High-to-low transitions on T2EX ignored.

1: High-to-low transitions on T2EX c

Bit2: TR2: Timer 2 Run Control.

This bit enables/disables Timer 2.

0: Timer 2 disabled.

1: Timer 2 enabled.

Bit1: C/T2: Counter/Timer Select.

0: Timer Function: Timer 2 incremented by clock defined by T2M (CKCON.5).

1: Counter Function: Timer 2 incremented by high-to-low transitions on external input pin (T2).

Bit0: CP/RL2: Capture/Reload Select.

This bit selects whether Timer 2 functions in capture or auto-reload mode. EXEN2 must be logic 1 for high-to-low transitions on T2EX to be recognized and used to trigger captures or reloads. If RCLK0

or TCLK0 is set, this bit is ignored and Timer 2 will function in auto-reload mode.

0: Auto-reload on Timer 2 overflow or high-to-low transition at T2EX (EXEN2 = 1).

RCLK0

0

0

X

1: Capture on high-to-low transition at T2EX (EXEN2 = 1).



16-bit Counter/Timer with Capture

Baud Rate Generator for UARTO

Baud Rate Generator for UART0

Baud Rate Generator for UARTO

16-bit Counter/Timer with Auto-Reload

Registre CKCON

Figure 22.1. CKCON: Clock Control Register

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
-	T4M	T2M	T1M	T0M	Reserved	Reserved	Reserved	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
								0x8E

Bit7: UNUSED. Read = 0b, Write = don't care.

Bit6: T4M: Timer 4 Clock Select.

> This bit controls the division of the system clock supplied to Timer 4. This bit is ignored when the timer is in baud rate generator mode or counter mode (i.e. C/T4 = 1).

0: Timer 4 uses the system clock divided by 12.

Ne pas modifier II. Timer 4 uses the system clock.

Bit5: T2M: Timer 2 Clock Select.

This bit controls the division of the system clock supplied to Timer 2. This bit is ignored when the

timer is in baud rate generator mode or counter mode (i.e. C/T2 = 1).

0: Timer 2 uses the system clock divided by 12.

1: Timer 2 uses the system clock.

Bit4: T1M: Timer 1 Clock Select.

This bit controls the division of the system clock supplied to Timer 1.

0: Timer 1 uses the system clock divided by 12.

Ne pas modifier Timer 1 uses the system clock.

Bit3: T0M: Timer 0 Clock Select.

This bit controls the division of the system clock supplied to Counter/Timer 0.

0: Counter/Timer uses the system clock divided by 12.

Ne pas modifier dounter/Timer uses the system clock.

Bits2-0: Reserved. Read = 000b, Must Write = 000.



Modifier « proprement » les registres de contrôle

Cas typique: modifications de quelques bits d'un registre Ne pas écraser les bits non concernés par votre configuration Exemple – Config de CKCON pour le Timer2

On veut mettre à zéro le bit 5 de CKCON

```
➤ CKCON = 0X00; Conséquence?

✓ CKCON &= \sim 0 \times 20; 1101 1111

✓ Équivalent à CKCON = CKCON & 0\timesDF;

✓ Ou CKCON &= \sim (1 << 5);
```



Ne pas s'appuyer sur les initialisations des registres Post-Reset

- Même si, après un Reset, aucun registre n'est initialisé de manière aléatoire.
- Même si, la documentation contient la valeur post-reset de chaque registre.

On prendra soin de faire abstraction de cette initialisation postreset.

Pourquoi?

- Exemple d'une fonction d'initialisation ré-appelée en cours de programme....
- Débogage facilité



Registres de rechargement RCAP2

Calcul du contenu des registres de pré-chargement.

Hypothèse: CLK SYSTEM de 22,1184 Mhz et période de 10mS souhaitée (Tick Time).

CLK Timer: CLK SYSTEM/12= 1,8432MHz, soit une période de 0,542535 µS Nombre d'incréments Timer nécessaires:

Tick Time / Timer Period = 18432 incréments soit 4800 en hexadécimal $(10000 \, \mu\text{S} \, / \, 0.542535 \, \mu\text{S} = 18432)$

Le timer compte de sa valeur de pré-chargement jusqu'à 0xFFFF, puis repasse à la valeur de pré-chargement.

La valeur de préchargement est donc de (0xFFFF+1) - 0x4800 = 0xB800

RCAP2L = 0X00

RCAP2H = OXB8

Et les registres TL2 et TH2? Quel contenu?



Autres registres?



A-t-on configuré tous les registres nécessaires pour la mise en place d'un Timer fonctionnant en auto rechargement et produisant une interruption toutes les N ms?

Mise en place des interruptions



Ne pas oublier de mettre en place les interruptions.....

Interrupt Source	Interrupt Vector	Priority Order	Pending Flag	Bit addressable?	Cleared by HW?	Enable Flag	Priority Control
Reset	0x0000	Тор	None	N/A	N/A	Always Enabled	Always Highest
External Interrupt 0 (/INT0)	0x0003	0	IE0 (TCON.1)	Y	Y	EX0 (IE.0)	PX0 (IP.0)
Timer 0 Overflow	0x000B	1	TF0 (TCON.5)	Y	Y	ET0 (IE.1)	PT0 (IP.1)
External Interrupt 1 (/INT1)	0x0013	2	IE1 (TCON.3)	Y	Y	EX1 (IE.2)	PX1 (IP.2)
Timer 1 Overflow	0x001B	3	TF1 (TCON.7)	Y	Y	ET1 (IE.3)	PT1 (IP.3)
UART0	0x0023	4	RI0 (SCON0.0)	Y		ES0 (IE.4)	PS0 (IP.4)
Timer 2 Overflow (or EXF2)	0x002B	5	TF2 (T2CON.7)	Y		ET2 (IE.5)	PT2 (IP.5)



La documentation manque de rigueur: Il y a 2 drapeaux susceptibles de provoquer une interruption Timer2: **TF2 et EXF2**



Mise en place des interruptions Validation

Figure 12.9. IE: Interrupt Enable

	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
	EA	IEGF0	ET2	ES0	ET1	EX1	ET0	EX0	00000000
_	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
							L	(bit addressable)	0xA8

Bit7: EA: Enable All Interrupts.

This bit globally enables/disables all interrupts. When set to '0', individual interrupt mask settings are

overridden.

0: Disable all interrupt sources.

1: Enable each interrupt according to its individual mask setting.

Bit6: IEGF0: General Purpose Flag 0.

This is a general purpose flag for use under software control.

Bit5: ET2: Enabler Timer 2 Interrupt.

This bit sets the masking of the Timer 2 interrupt.

0: Disable Timer 2 interrupt.

1: Enable interrupt requests generated by the TF2 flag (T2CON.7).

Bit4: ES0: Enable UART0 Interrupt.

This bit sets the masking of the UART0 interrupt.

0: Disable UART0 interrupt.

1: Enable UART0 interrupt.





Mise en place des interruptions - Priorité

Figure 12.10. IP: Interrupt Priority

R/W	Reset Value							
-	-	PT2	PS0	PT1	PX1	PT0	PX0	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
						L	(bit addressable)	0xB8

Bits7-6: UNUSED. Read = 11b, Write = don't care.

Bit5: PT2: Timer 2 Interrupt Priority Control.

This bit sets the priority of the Timer 2 interrupt.

0: Timer 2 interrupt priority determined by default priority order.

1: Timer 2 interrupts set to high priority level.

Bit4: PS0: UART0 Interrupt Priority Control.

This bit sets the priority of the UART0 interrupt.

0: UART0 interrupt priority determined by default priority order.

1: UART0 interrupts set to high priority level.

Code Config_Timer2_BT

Fonction: CFG_Timer2_BT

De préférence dans l'ordre:

- Stopper le Timer2 (au cas où) Action sur TR2 de T2CON
- Configurer CKCON pour positionner T2M: Timer 2 Clock Select
 CLK Timer2 = sysclk/12
- Configuration T2CON
 - Effacer le flag d'overflow TF2
 - Effacer le flag d'overflow EXF2 (pour éviter une interruption intempestive)
 - Configurer le mode 1 (action sur CPRL2, RCLK0 et TCLK0)
 - Fonctionnement en mode Timer Action sur C/T2
 - Dévalider l'action possible de l'entrée T2EX Action sur EXEN2
- Programmation des registres de rechargement RCAP2L et RCAP2H
- Initialisation des registres de comptage du Timer T2L et T2H
- Configuration de la priorité pour interruption Timer2 action sur le registre IP
- Validation de l'interruption Timer2- action sur le registre IE
- Timer 2 démarré Action sur T2CON

Et la validation globale des interruptions ? (bit EA)



Code ISR Timer2

```
void ISR_Timer2(void) interrupt 5
{
    Visu_INT_Timer2 = 1;  // Facultatif -
Visualisation matérielle de l'interruption

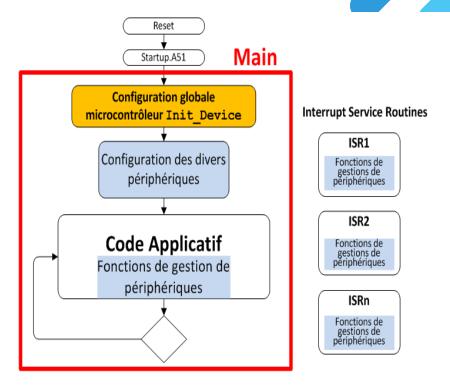
    TF2 = 0;  // RAZ Flag Timer2

    // Insertion du code de l'application

    Visu_INT_Timer2 = 0;
}

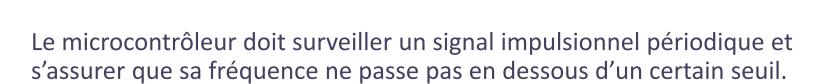
Que se passe t-il si l'on omet
```

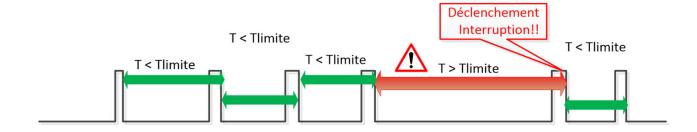
 \ll TF2 = 0 \gg ?





Pause Timer2 Auto-rechargement...





Comment à l'aide du Timer 2, configuré en mode auto-rechargement, peut-on générer une interruption lorsque la durée entre 2 impulsions dépasse la durée limite?



