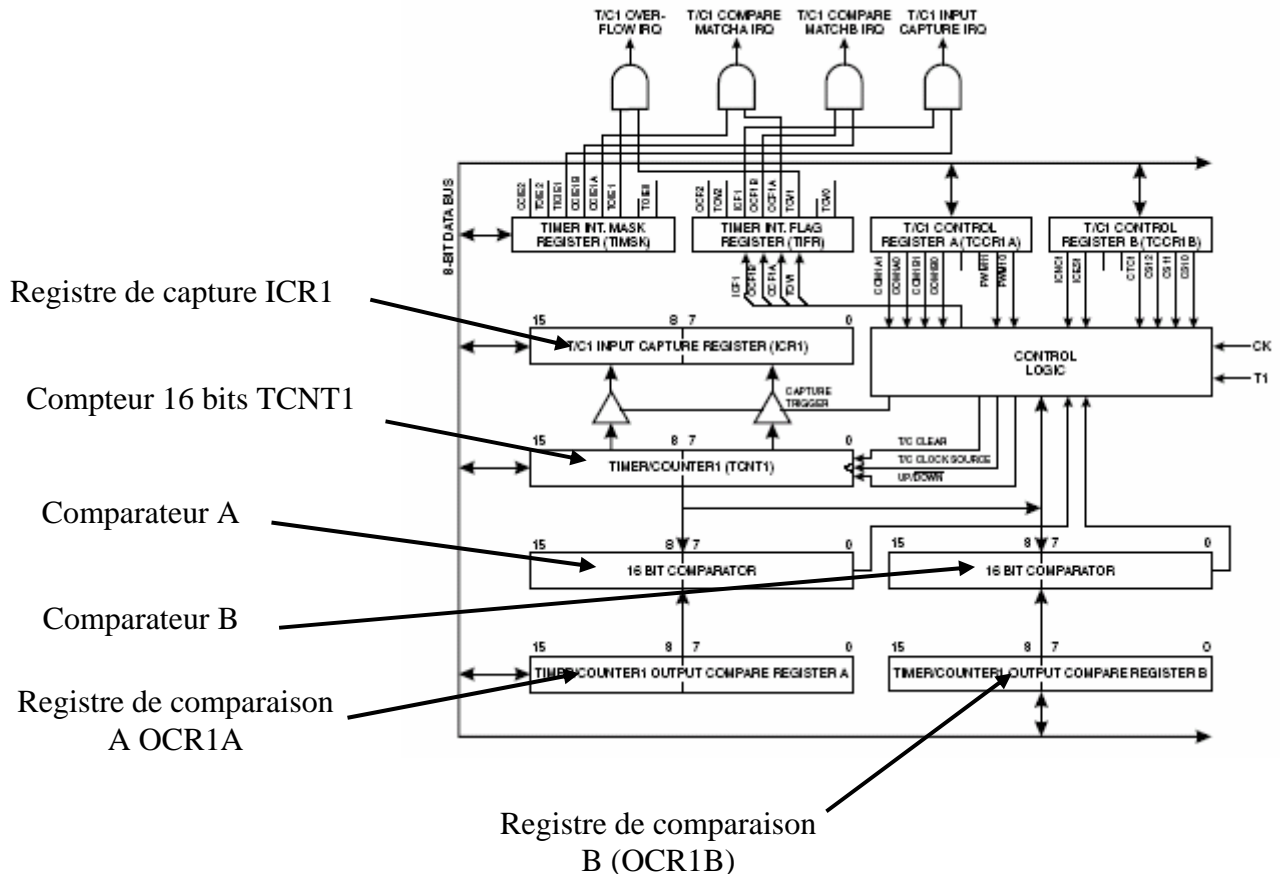


TD n°3 : Mise en oeuvre des compteurs/temporisateurs du AT90S8535

Objectifs du TD:

- Configurer les registres d'initialisation du timer 1 pour générer une interruption périodique de 400 ms. A chaque interruption on viendra mettre alternativement à 1 ou à 0 le bit 0 du port B. (On utilisera le compteur en mode comptage jusqu'à la valeur contenue dans OCR1A puis remise à 0 après comparaison réussie).
- Mise en œuvre du compteur 2 en mode PWM (Pulse Width Modulation).

Synoptique du compteur/temporisateur 1 :



Voir la description détaillée des registres dans l'annexe du polycopié.

- 1) On justifiera l'initialisation à 0 du registre tccr1a et à \$0C celle du registre tccr1b.
- 2) Interruption toutes les 400ms :

On divise l'horloge de base par 256 => 4 Mhz / 256 = 15625 Khz ce qui fait une période d'horloge du compteur de 64 μ s.

Détermination de la valeur N de préchargement du registre OCR1A :

$$N = 400\text{ms} / 64 \mu\text{s} = 6250 \text{ soit } \$186A$$

```

"8535def.inc"
.def temp = r16
.cseg
.org $000
    rjmp    init           ;Reset interrupt
.org $006
    rjmp    timer1_compA
.org $015
init:
    ldi     temp,low (RAMEND)
    out     SPL,temp       ; initialisation pointeur pile Pf
    ldi     temp,high (RAMEND)
    out     SPH,temp       ; initialisation pointeur pile PF
    ser     temp
    out     ddrb, temp     ; port b en sortie
    out     portb, temp    ; sorties portb à 1
    rcall   init_timer1   ;initialisation timer1
    sei                    ;valide les interruptions

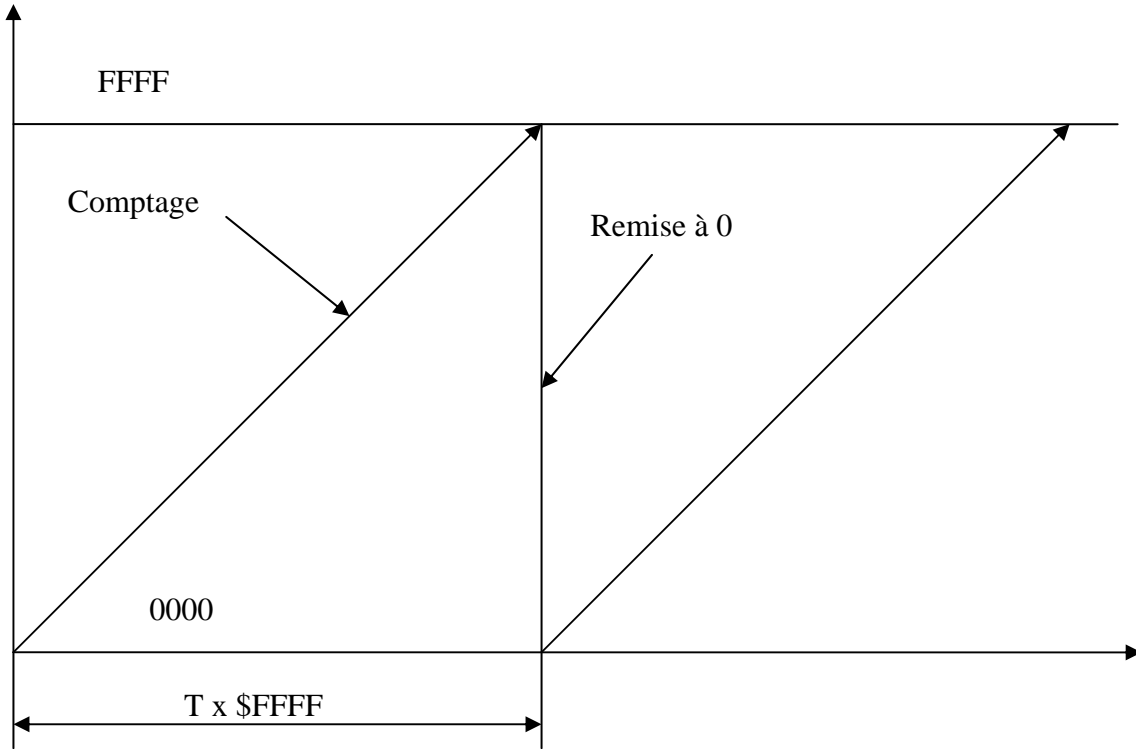
;-----
; programme principal
;-----
debut:
    nop
    nop
    nop
    rjmp    debut

;-----
init_timer1:
    ldi     temp, $0C
    out     tccr1b, temp   ; clk/256 soit 64µs
    ldi     temp, $00
    out     tccr1a, temp   ; timer1 en mode normal
    ldi     temp, $18
    out     ocr1ah, temp   ; registre ocr1A
    ldi     temp, $6A
    out     ocr1al, temp   ; initialisé à 6250
    ldi     temp, $10
    out     tmsk, temp     ; soit $186A (it toutes 400ms)
    out     tmsk, temp     ; autorise l'it ocr1A
    ret

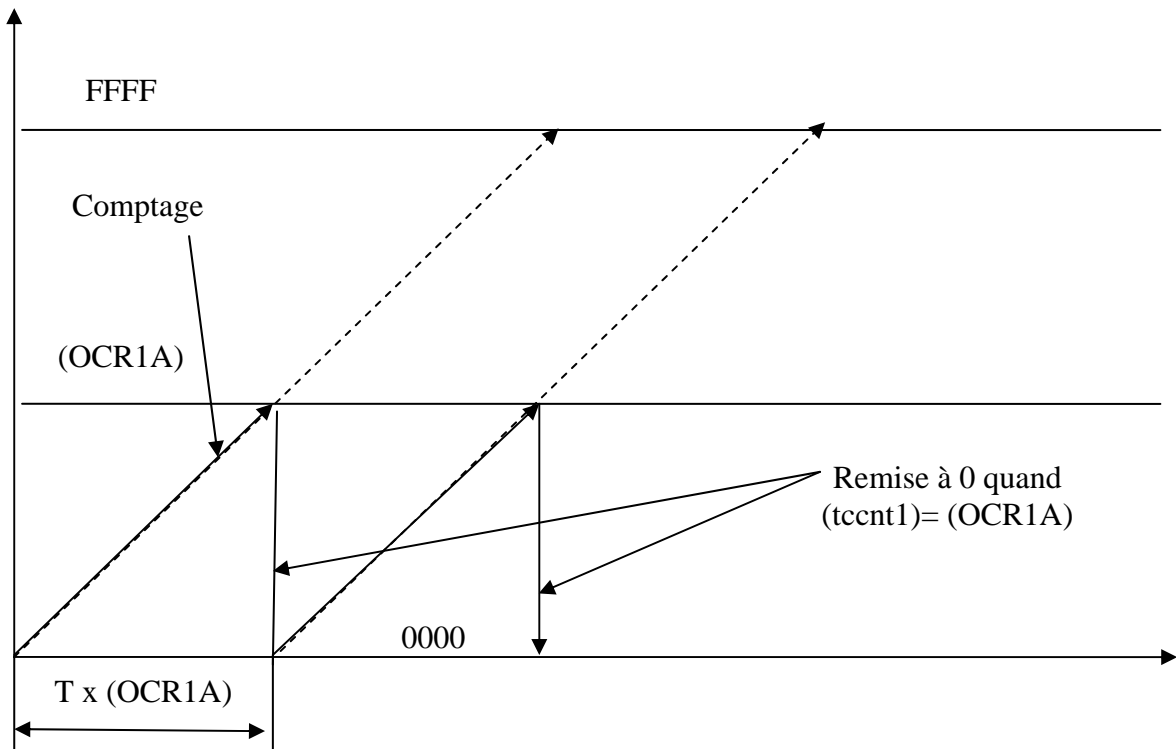
;-----
; sous-prog de traitement de l'interruption
;-----
timer1_compA:
    sbis    portb, 0       ;test bit 0 du portb
    rjmp    m_a_1_b0      ;si 0 met à 1
    rjmp    m_a_0_b0      ;si 1 met à 0
m_a_1_b0:
    sbi     portb, 0
    rjmp    fin_a
m_a_0_b0:
    cbi     portb, 0
fin_a:
    reti

```

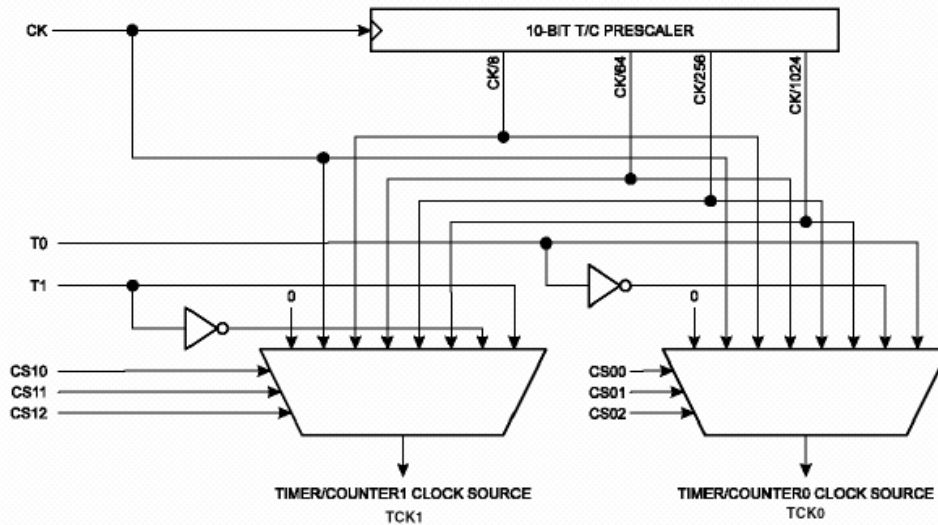
Principe de fonctionnement du timer 1 en mode continu :



Principe de fonctionnement du timer 1 en mode réinitialisation :



Le module de prédivision :



Les registres du timer1 :

- TCCR1A et TCCR1B** (Timer Counter Control Register 1A et 1B)
- TCNT1** (Timer CouNter 1)
- OCR1A et OCR1B** (Output Compare Register 1A et 1B)
- ICR1** (Input Capture Register 1)
- TIFR** (Timer Interrupt Flag Register) : commun aux 3 Timers
- TIMSK** (Timer Interrupt MaSK) : commun aux 3 Timers

TCCR1A : 1^{er} registre de contrôle du timer 1.

Adresse	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
\$2F (\$4F)	COM1A1	COM1A0	COM1B1	COM1B0	-----	-----	PWM11	PWM10

COM1X1	COM1X0	Action programmée sur OC1X
0	0	Timer 1 déconnecté (pas d'action)
0	1	Complémentation du niveau
1	0	Mise à 0
1	1	Mise à 1

Les bits COMX1 et COMX0 permettent de définir l'action qui doit se produire sur la sortie OC1X lorsque les valeurs présentes dans le registre de comparaison OCR1X et le compteur TCNT1 sont égales. Voir table ci-contre :

« X » est à remplacer par A pour la sortie OC1A et par B pour la sortie OC1B.

TCCR1B : 2^{ème} registre de contrôle du timer 1.

Adresse	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
\$2E (\$4E)	ICNC1	ICES1	-----	-----	CTC1	CS12	CS11	CS10

CS12	CS11	CS10	Source d'horloge du compteur
0	0	0	Aucune : Le compteur est stoppé
0	0	1	Horloge système (CK)
0	1	0	Horloge système (CK) / 8
0	1	1	Horloge système (CK) / 64
1	0	0	Horloge système (CK) / 256
1	0	1	Horloge système (CK) / 1024
1	1	0	Broche T1, active sur front descendant
1	1	1	Broche T1, active sur front montant

TCNT1 : Compteur libre du timer 1 (16 bits), accessible en lecture et en écriture

Adresse	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
\$2D (\$4D)	MSB								TCNT1H TCNT1L
\$2C (\$4C)								LSB	

OCR1A et OCR1B : Registres de comparaison du timer 1 (16 bits), accessibles en lecture et en écriture

Adresse	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
\$2B (\$4B)	MSB								OCR1AH OCR1AL
\$2A (\$4A)								LSB	
Adresse	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
\$29 (\$49)	MSB								OCR1BH OCR1BL
\$28 (\$48)								LSB	

ICR1 : Registre de capture du timer 1 (16 bits), accessible en lecture

Adresse	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
\$27 (\$47)	MSB								ICR1H ICR1L
\$26 (\$46)								LSB	

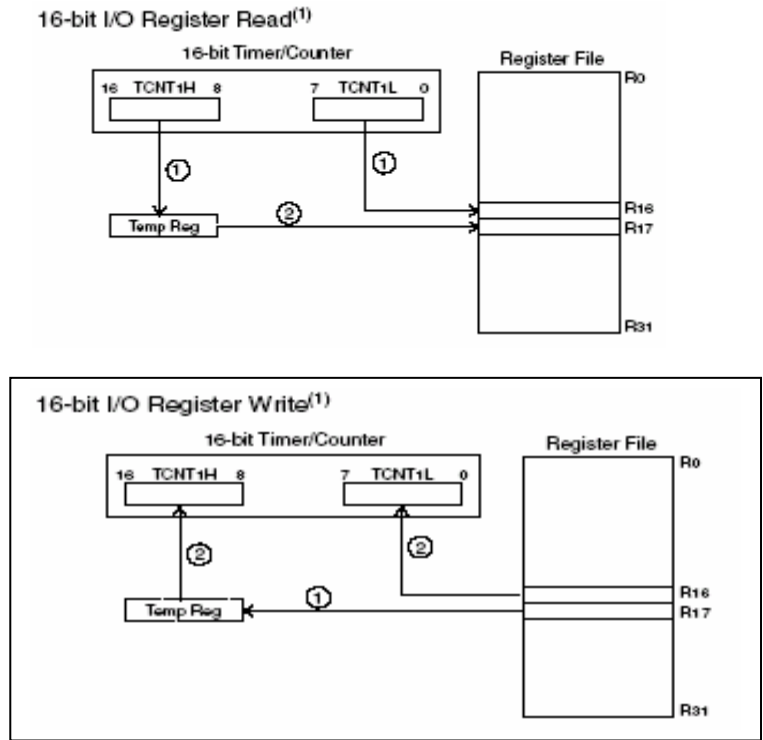
TIFR : Registre d'état des timers, indique l'état des compteurs et des comparaisons.

Adresse	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
\$38 (\$58)	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	-----	TOV0

- TOV1 : Ce bit passe à 1 lors du débordement du compteur du Timer1 , TCNT1.
- OCF1A : Ce bit passe à 1 lorsque TCNT1 = OCR1A (Comparaison A réussie).
- OCF1B : Ce bit passe à 1 lorsque TCNT1 = OCR1B (Comparaison B réussie).
- ICF1 : Ce bit passe à 1 lorsqu'un front actif se produit sur l'entrée ICP.

TIMSK : Registre de validation des interruptions des timers, permet de valider la génération d'interruptions par le timer .(A condition que le bit I du registre SREG soit à 1)

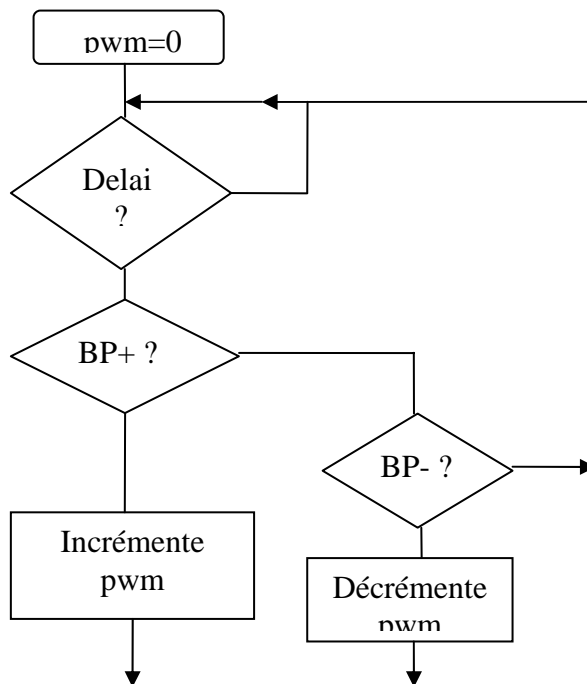
Adresse	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
\$39 (\$59)	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	-----	TOIE0



Procédure d'accès en lecture/écriture dans un registre de 16 bits.

3) Mise en œuvre du timer 2 en mode PWM :

On désire réaliser la mise en route et l'arrêt progressif d'un moteur à courant continu. Pour cela on utilise le timer 2 en mode PWM comme signal de commande et deux boutons poussoirs BP+ et BP- pour accélérer ou ralentir la vitesse du moteur. Le fonctionnement de l'ensemble est représenté par l'algorithme ci-dessous. Proposer un programme assembleur qui réalise cette fonction.



```

;-----
; Ce programme lit les BP +/- toutes les 40ms
; si BP+ maintenu appuyé on augmente le Rapport
; Cyclique RC de la PWM jusqu'au max
; si BP- maintenu appuyé on diminue le Rapport
; Cyclique RC de la PWM jusqu'au min
; ce programme utilise la lère partie du TP
; BP+ et BP- câblés portc bit 0 et 1
;-----
.include "8535def.inc"
.def temp = r16
.cseg
.org $000
    rjmp init          ;Reset interrupt
.org $015
init:
    ldi    temp,low (RAMEND)
    out    SPL,temp          ; initialisation pp Pf
    ldi    temp,high (RAMEND)
    out    SPH,temp          ; initialisation pp PF
    ser    temp
    out    ddrd, temp        ; port d en sortie
    clr    temp
    out    ddrc, temp        ; portc en entrée
    rcall  init_timer2
    rjmp  debut

;-----
debut:
    rcall  delai_40ms
    in    temp, pinc          ; lecture portc
    sbrc  temp, 0            ; test BP+
    rjmp  pwm_plus           ; si BP+ appuyé on va à pwm_plus
    sbrc  temp, 1            ; test BP-
    rjmp  pwm_moins          ; si BP- appuyé on va à pwm_moins
    rjmp  fin

pwm_plus:
    in    temp, ocr2
    cpi   temp, $FF          ;test si RC au max
    breq  fin
    inc   temp                ;on incrémente le RC
    out   ocr2, temp
    rjmp  fin

pwm_moins:
    in    temp, ocr2
    cpi   temp, 0            ;test si RC au min
    breq  fin
    dec   temp                ;on décrémente le RC
    out   ocr2, temp

fin:
    rjmp  debut

;-----
;-----
;    init_timer2: initialise le timer2 en mode PWM
;    rapport cyclique nul par défaut
;    sortie pin n°21 (PD7)
;-----
init_timer2:
    ldi   temp, $62
    out   tccr2, temp
    clr   temp
    out   ocr2, temp
    ret

;-----

```