

# Le timer1 de l'ATMEL ATmega16

## 1 Présentation

'Multi function Timer' 16 bits fonctionnant en modes

- Overflow
- Comparaison sur 2 voies permettant 3 configurations :
  1. Mode CTC (clear timer on compare match)
  2. Mode Fast PWM
  3. Mode Fast correct PWM
- Compteur d'événements
- Générateur de fréquence
- Capture sur 1 voie

Registre de comptage du Timer1 : Compteur/décompteur 16 bits accessible sous forme de 2 octets : **TCNT1H** (R/W) et **TCNT1L** (R/W) ou d'un int **TCNT1** (R/W).

Interruptions possible sur overflow (1), comparaison (2) et capture (1).

Sortie sur comparaison OC1A (Pd5), OC1B (Pd4), entrée de capture ICP1 (Pd6) entrée de comptage externe T1 (Pb1).

## 2 prédivision

### TCCR1B: Timer1 control register B

7	6	5	4	3	2	1	0
ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10

CS12	CS11	CS10	Description
0	0	0	Stop timer0
0	0	1	CK
0	1	0	CK/8
0	1	1	CK/64
1	0	0	CK/256
1	0	1	CK/1024
1	1	0	Pin T1 ↓ (PB1)
1	1	1	Pin T1 ↑ (PB1)

CK est l'horloge de la CPU (la fréquence du quartz).

Si T1 est utilisée, Pb1 sera configurée en entrée.

! Une écriture dans TCNT1 doit toujours commencer par TCNT1H avant TCNT1L ; masquer les IT durant cette opération.

**Le comptage maxi est donc  $65535 \cdot 1024 = 67\ 107\ 840$  Tquartz.**

Ex : Avec un quartz de 7.3728MHz, la durée correspondante est supérieure à 9.1s.

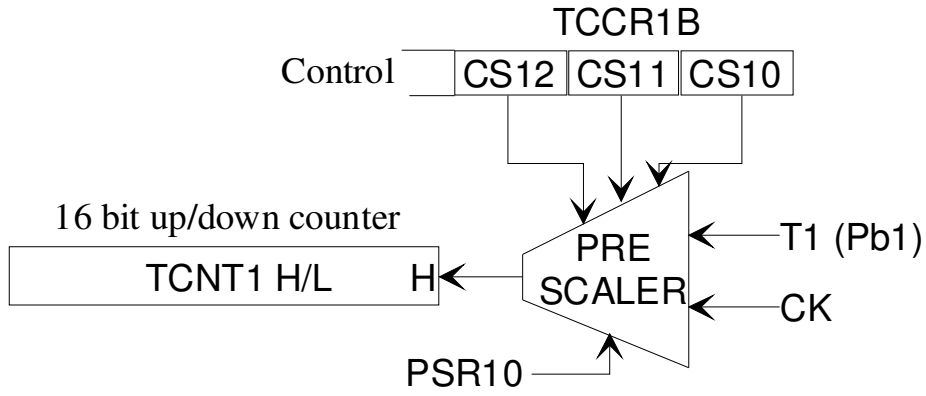
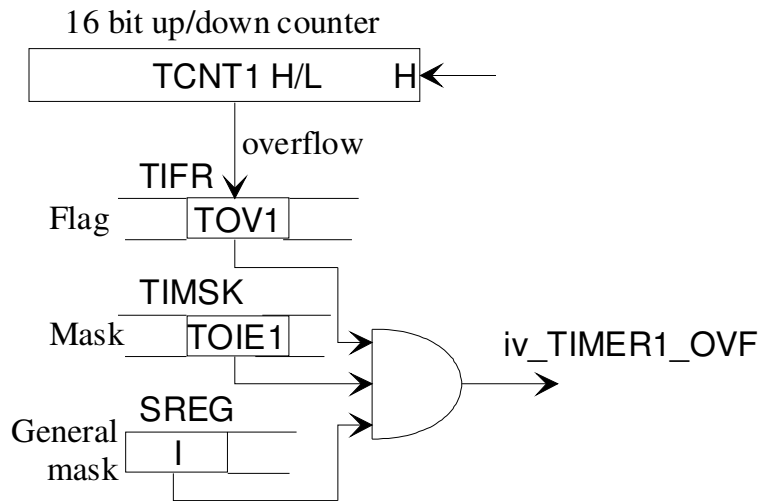


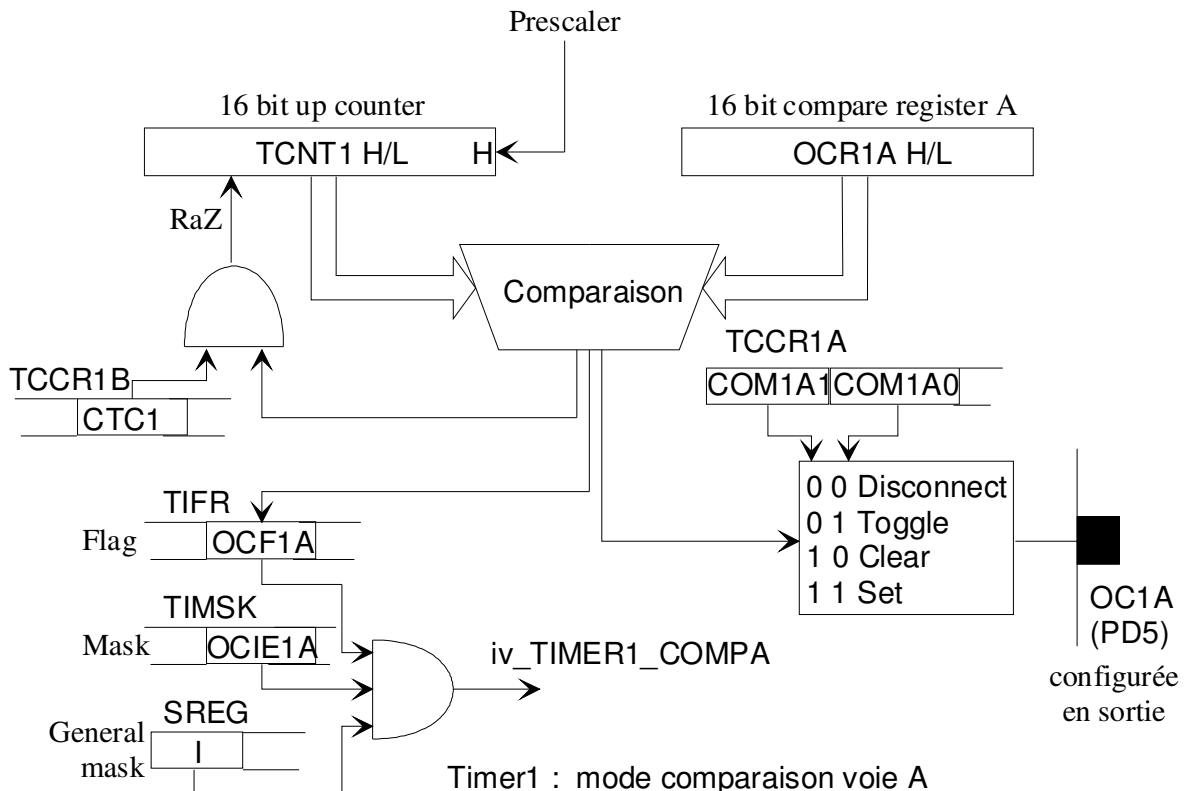
Schéma équivalent du prédiviseur du Timer1

### 3 mode overflow du timer1



Mécanisme d'interruption sur overflow du Timer1

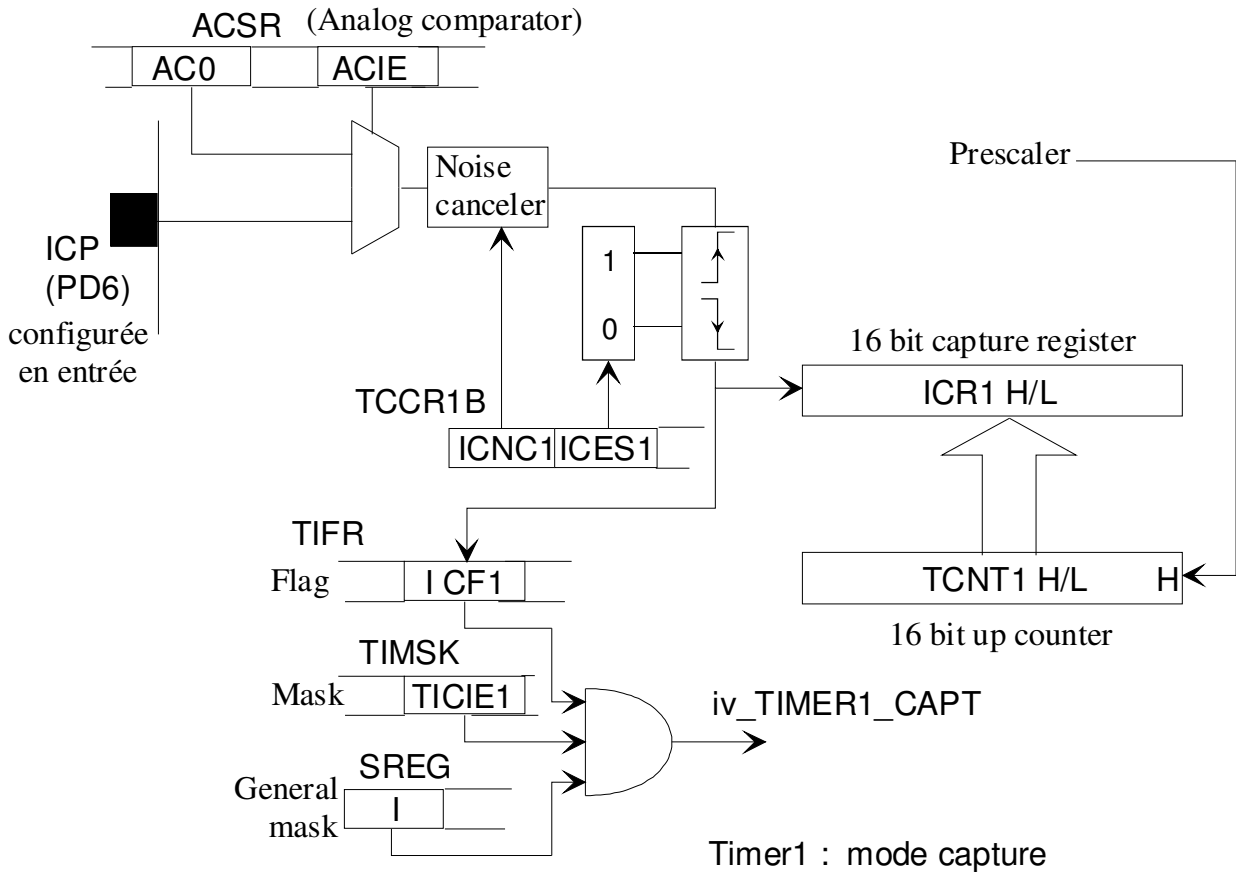
### 4 Mode comparaison du timer1 (voie A)



Timer1 : mode comparaison voie A

OCR1AH , OCR1AL et OCR1BH, OCR1BL : registres de comparaison voies A & B.  
**! Une écriture dans OCR1A doit toujours commencer par OCR1AH avant OCR1AL ; masquer les IT durant cette opération. Idem pour OCR1B**  
 On peut accéder à ces registres sous forme de int : **OCR1A** et **OCR1B**

## 5 Mode capture du timer1



**ICR1H** et **ICR1L** : Registres de capture, accessibles sous forme de int : **ICR1**.

**TCCR1B** : Timer1 control register B

7	6	5	4	3	2	1	0
<b>ICNC1</b>	<b>ICES1</b>	-	WGM13	WGM12	CS12	CS11	CS10

**ICNC1** : input capture noise canceler

**ICES1** : input capture edge select

Si le filtrage de l'entrée de capture est activé (ICNC1 = 1), le flag de capture ICF1 est mis à 1 après que 4 échantillonnages successifs de l'entrée ICP aient donné le même niveau logique correspondant à celui programmé pour la capture par ICES1.

**! Une lecture de ICR1 doit toujours commencer par ICR1L puis ICR1H ; masquer les IT durant cette opération.**

## 6 Les modes de fonctionnement du timer1

### TCCR1A et TCCR1B : Timer1 control register A et B

7	6	5	4	3	2	1	0
COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10

7	6	5	4	3	2	1	0
ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10

Ces modes sont semblables à ceux du timer0, mais sur 16 bits et déclinés en 16 variantes :

Mode	WGM13	WGM12	WGM11	WGM10	#	TOP
Normal	0	0	0	0	0	0xFFFF
Phase correct PWM 8 bits	0	0	0	1	1	0x00FF
Phase correct PWM 9 bits	0	0	1	0	2	0x01FF
Phase correct PWM 10 bits	0	0	1	1	3	0x03FF
CTC	0	1	0	0	4	OCR1A
Fast PWM 8 bits	0	1	0	1	5	0x00FF
Fast PWM 9 bits	0	1	1	0	6	0x01FF
Fast PWM 10 bits	0	1	1	1	7	0x03FF
PWM, phase & frequency correct	1	0	0	0	8	ICR1
PWM, phase & frequency correct	1	0	0	1	9	OCR1A
Phase correct PWM	1	0	1	0	10	ICR1
Phase correct PWM	1	0	1	1	11	OCR1A
CTC	1	1	0	0	12	ICR1
Reserved	1	1	0	1	13	-
Fast PWM	1	1	1	0	14	ICR1
Fast PWM	1	1	1	1	15	OCR1A

### 6.1 Description des modes et évolution des sorties correspondantes

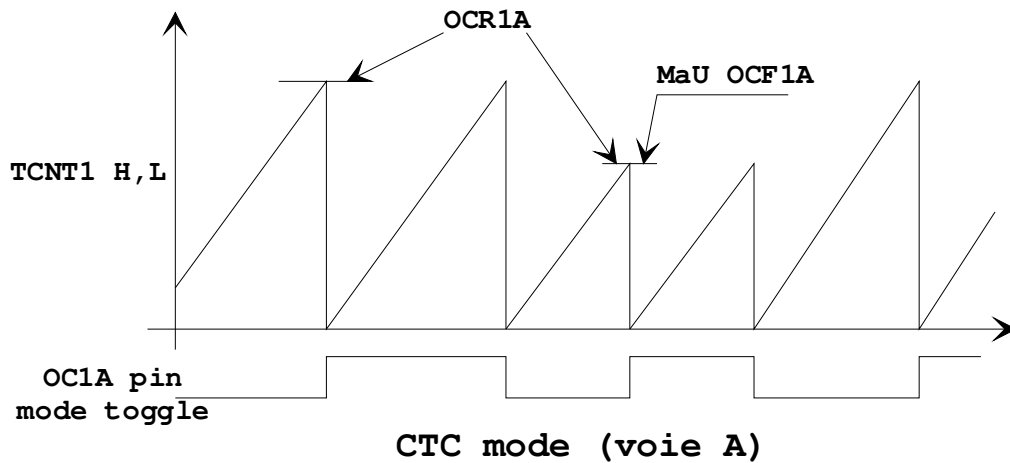
TCCR1A : Timer1 control register A

7	6	5	4	3	2	1	0
COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10

#### 6.1.1 Mode comparaison non PWM (normal ou CTC modes)

COM1A1/COM1B1	COM1A0/COM1B0	Description
0	0	Timer1 déconnecté de la sortie OC1A/OC1B
0	1	Toggle OC1A/OC1B
1	0	RaZ OC1A/OC1B
1	1	MaU OC1A/OC1B

Exemple : Clear on compare sur voie A avec OC1A en mode toggle.



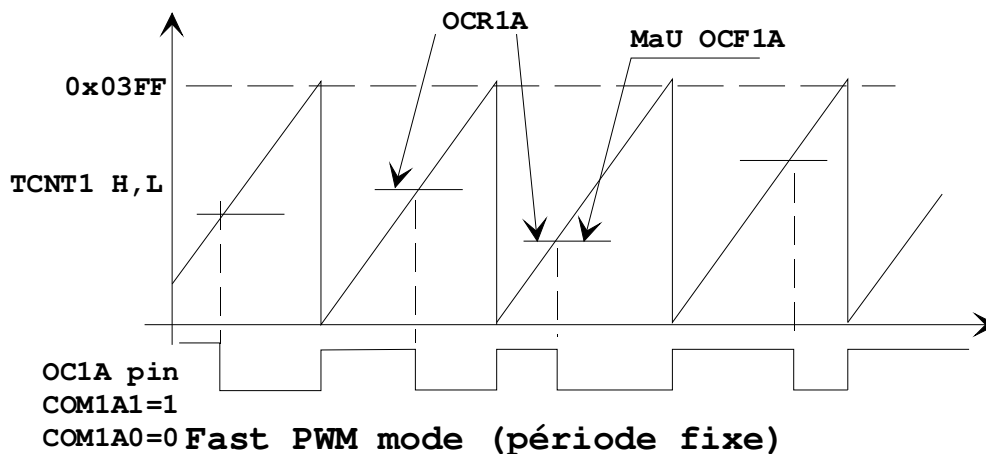
### 6.1.2 Mode Fast PWM

La valeur MAX peut être, selon le mode : 0x00FF, 0x01FF, 0x03FF, OCR1A, ICR1

COM1A1/COM1B1	COM1A0/COM1B0	Description
0	0	Timer1 déconnecté de la sortie OC1A/OC1B
0	1	Toggle OC1A, OC1B déconnecté
1	0	RaZ OC1A/OC1B sur comparaison, MaU OC1A, OC1B sur MAX
1	1	MaU OC1A/OC1B sur comparaison, RaZ OC1A, OC1B sur MAX

En utilisant OCR1A comme valeur MAX et OCR1B pour la comparaison, on peut générer un signal modulé à la fois en fréquence et en rapport cyclique, sur la sortie OC1B.

Exemple : signal Fast PWM de période fixe sur sortie OC1A.

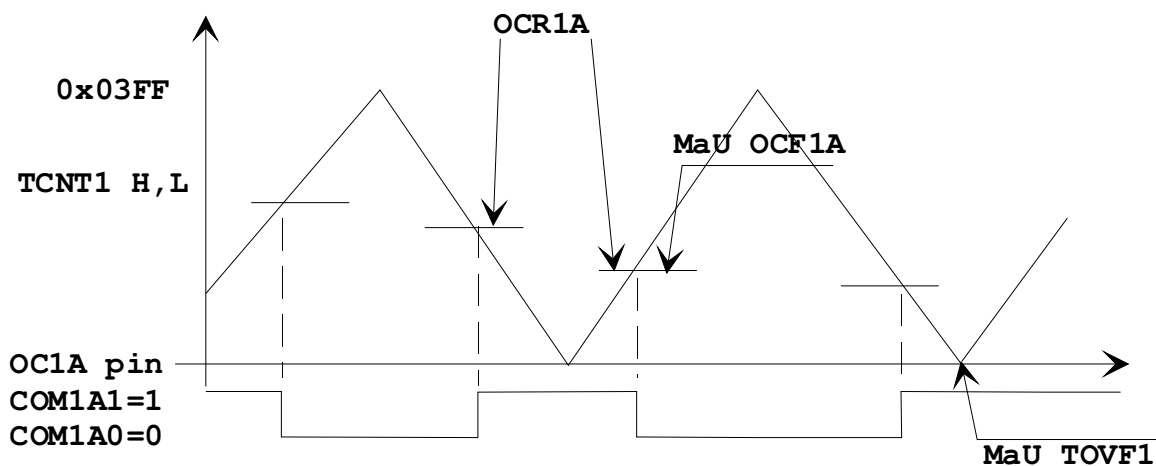


### 6.1.3 Modes Phase correct et Phase & frequency correct

Dans le mode phase correct, la période est fixe, 0x00FF, 0x01FF ou 0x03FF.

Dans le mode phase & frequency correct, la période est OCR1A ou ICR1

COM1A1/COM1B1	COM1A0/COM1B0	Description
0	0	Timer1 déconnecté de la sortie OC1A/OC1B
0	1	Toggle OC1A, OC1B déconnecté pour modes 13 et 15 sinon OC1A également déconnecté
1	0	RaZ OC1A/OC1B sur comparaison lors du comptage, MaU OC1A, OC1B sur comparaison lors du décomptage,
1	1	MaU OC1A/OC1B sur comparaison lors du comptage, RaZ OC1A, OC1B sur comparaison lors du décomptage,



**Fast correct PWM mode (période fixe)**

```

/*****
/*          Timer1 en mode phase correct PWM sans IT          */
/*****
// ATmega 16 (simu OK)
// le 18 09 03, programme simple de test
// Signal carré,
// Période fixe TCNT1 max = 0x00FF, Prédiv 1
// OCR1 = 0x80
// signal disponible sur OC1A (Pd5)

#include <iom16v.h>
#define PULSE 128

// Initialisation du timer1 : Phase correct PWM_0xFF
void Init_T1_Phase_correct_PWM (void)
{
    TCCR1B = (1<<CS10);           // Prédivision 1
    TCCR1A = (1<<WGM10);         // Mode #1
    TCCR1A |= (1<<COM1A0)+(1<<COM1A1);
                                // MaU OC0 sur overflow

```

```

        DDRD = (1<<5)                // OC1A en sortie
        OCR1H = 0;                   // chargement initial
        OCR1L =PULSE;
    }

// Programme principal
void main (void)
{
    Init_T1_Phase_correct_PWM ();
    while (1)
        {
        }
}

```

## 7 Les registres de commande et d'état du timer1

### 7.1 TIFR : Timers interrupt flag register

7	6	5	4	3	2	1	0
OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	-	TOV0

- ICF1 :** Timer1 input capture flag, mise à '0' automatique avec l'IT correspondante ), ou en écrivant un '1' dans ICF1.
- OCF1A :** Timer1 output compareA flag, mise à '0' automatique avec l'IT correspondante ), ou en écrivant un '1' dans OCF1A.
- OCF1B :** Timer1 output compareB flag, mise à '0' automatique avec l'IT correspondante ), ou en écrivant un '1' dans OCF1B.
- TOV1 :** Timer1 overflow flag, mise à '0' automatique avec l'IT correspondante (#10), ou en écrivant un '1' dans TOV1.

### 7.2 TIMSK : Timers interrupt mask register

7	6	5	4	3	2	1	0
OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	-	TOIE0

- TICIE1 :** Timer1 input capture interrupt mask
- OCIE1A :** Timer1 output compareA interrupt mask
- OCIE1B :** Timer1 output compareB interrupt mask
- TOIE1 :** Timer1 overflow interrupt mask

Lorsque l'E/S Pd5 est utilisée par le timer (OC1A), elle doit être configurée en sortie.

Lorsque l'E/S Pd4 est utilisée par le timer (OC1B), elle doit être configurée en sortie.

Lorsque l'E/S Pd6 est utilisée par le timer (ICP), elle doit être configurée en entrée.

### 7.3 TCCR1A : Timer1 control register A

7	6	5	4	3	2	1	0
COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10

**FOC1A :** Force output compare 1A. En écrivant un '1' dans FOC1A, on fait évoluer le niveau logique de OC1A selon la configuration de COM1A1 et COM1A0, indépendamment de la comparaison.

**FOC1B :** Force output compare 1B. En écrivant un '1' dans FOC1B, on fait évoluer le niveau logique de OC1B selon la configuration de COM1B1 et COM1B0, indépendamment de la comparaison.

#### 7.4 TCCR1B : Timer1 control register B

7	6	5	4	3	2	1	0
ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10

#### 7.5 SFIOR : Special function IO register

(registre commun aux 3 timers de l'ATMega 16)

7	6	5	4	3	2	1	0
ATDS2	ATDS1	ATDS0	-	ACME	PUD	PSR2	<b>PSR10</b>

**PSR10 :** Prescaler reset Timer1 et Timer0. Le prédiviseur des timer0 et 1 est mis à '0' lorsque ce bit est mis à '1', il est ensuite automatiquement remis à '0'.