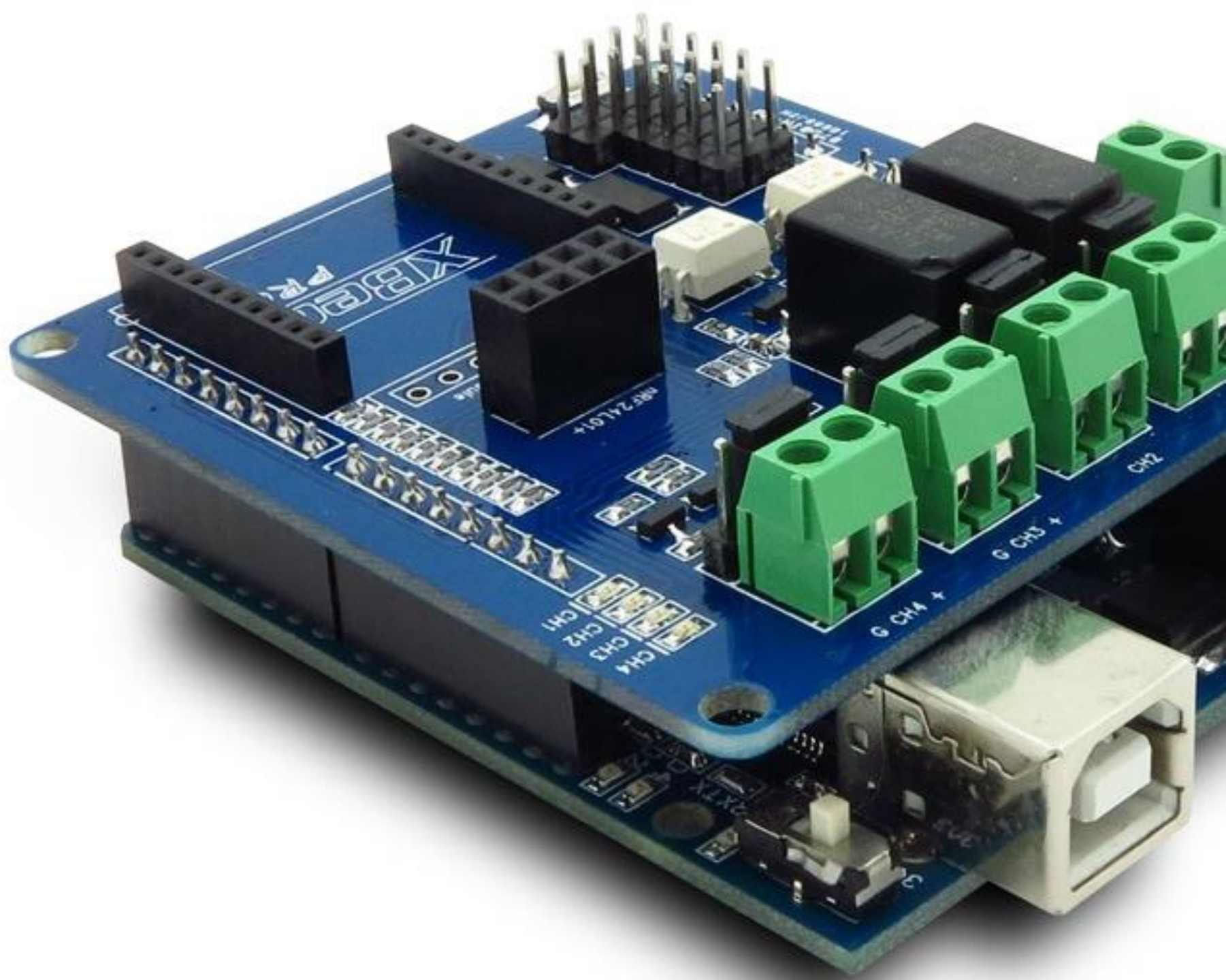


TIMERS

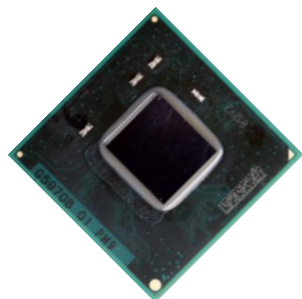
SISTEMAS EMBEBIDOS





TimerOne

Esta biblioteca es un conjunto de rutinas para configurar el temporizador de hardware de 16 bits llamado Timer1 en el ATmega168 / 328.





TimerOne

Teóricamente podríamos fijar una interrupción cada $1/16000000$ segundos, lo que no sería muy útil porque cada instrucción de Arduino necesita varios pulsos de reloj para ejecutarse (y algunos muchos pulsos).

Por eso cada Timer dispone de un registro interno que indica cada cuantos ticks del reloj debe dispararse. Básicamente el que nos interesa es un Timer cuyo registro es un entero sin signo de 16 bits, lo que le permite contar hasta $2^{16} = 65.536$

Si este contador se disparase cada tick del cristal, no nos serviría de mucho porque, el máximo tiempo que el contador podría abarcar sería de:



$$\frac{1}{16.000.000} 65.536 = 0,004096 \text{ segs} = 4, \text{ milisegundos}$$

TimerOne.h

Timer1.initialize

Timer1.attachInterrupt

#include <TimerOne.h> libreria

Timer1.initialize(1000000);

1 segundo según prescaler y reloj

Timer1.pwm(pin,dutty);

Pin 9 o 10, dutty entre 0 y 1023

Timer1.attachInterrupt(subrutina);

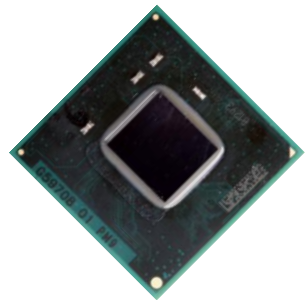
Interrupción de timer

Subrutina:

Sintaxis:

void subrutina () {

}



Código Timer1.ino

```
#include <TimerOne.h>
int i=0;
void setup() {
  pinMode(9,OUTPUT);
  Timer1.initialize(1000000);
  Timer1.attachInterrupt(conteo);
  Serial.begin(9600);
}
```

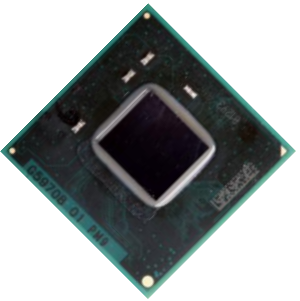
Librería de timer1

```
void loop() {
```

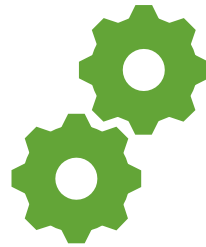
Configuración básica de timer

```
  }
  void conteo () {
    i++;
    Serial.println(i);
    if(i==10){
      digitalWrite(9, digitalRead(9) ^ 1);
      i=0;
    }
  }
}
```

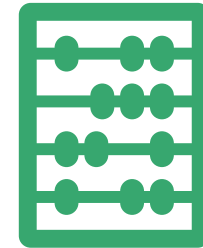
Subrutina de timer1



PROGRAMACIÓN



CÓDIGO 17: MANEJO DE
TIMER 1



CÓDIGO 18: MANEJO DE
TIMER 2