

```

/* ATB_B1_Timer_1.c Created: 25.11.2014 19:23:07 Author: AS */

#define F_CPU 16000000UL // Angabe der Quarzfrequenz, wichtig für die Zeit
#include <avr/io.h> // Einbindung Datei Ausgänge
#include <avr/interrupt.h>
volatile int8_t flag_1ms; // Globale Variable flag_1ms
volatile int16_t warten; // Globale Variable warten

void timer_init() // Timer 0 konfigurieren
{
    TCCR0A = 0; // Es werden keine Bits gesetzt
    TCCR0B = (1<<WGM01)|(1<<CS01)|(1<<CS00); // Einstellung CTC Modus, Prescaler 64
    TCNT0=1; // Initialisiert Timer
    OCROA=249; // Laden des Vergleichswertes
    TIMSK0|=(1<<OCIE0A); // Interrupt erlauben
}

ISR (TIMER0_COMPA_vect) // ISR
{
    flag_1ms=1; // setzt flag_1ms auf 1
}

int main(void)
{
    timer_init(); // Initiiert Timer (erster Aufruf)
    DDRC=0b01000000; // setzt Port C, Pin PC6 auf Ausgang
    sei(); // gibt Interrupts frei
    while(1) // Beginn Programmschleife while
    {
        if(flag_1ms) // Abfrage ob flag_1ms wahr (1) ist
        { // Wenn Abfrage wahr ist, dann ...
            flag_1ms=0; // setzt flag_1ms auf 0
            warten++;
            if(warten==500) // Angabe Zeit 500ms
            {
                PORTC &= ~(1<<PC6); // Schaltet PC6 ein
            }
        }
        else
        {
            if(warten==1000) // Angabe Zeit 500 ms
            {
                PORTC |= (1<<PC6); // Schaltet PC6 aus
                warten=0; // setzt Zled7 auf 0
            }
        }
    } // ende if warten
} // ende if flag
} // ende while
} // ende main

```