

```
#include <avr\io.h>
#include <avr\interrupt.h>
#define hiCnt 1136/2
#define loCnt 2273/2
unsigned delay;
void dlyby100msOC(int k);
void main (void)
{
    DDRB |= 0x20; // configure OC1A/PB5 for output
    delay = loCnt; // prepare to generate the low tone (440 Hz)
    TCCR1A = 0x40; // configure OC1A to normal mode, compare match action set to toggle
    TCCR1B = 0x02; // use clk_I/0 / 8 as Timer 1 clock source
    OCR1A = TCNT1 + delay; // start the first compare operation
    TIMSK1 = 0x02; // enable OC1A compare match interrupt
    TIFR1 = 0x02; // clear OCF1A flag
    sei(); // enable interrupt globally
    while (1) {
        dlyby100msOC(5); // wait for 0.5 s
        delay = hiCnt; // switch to delay count for 880-Hz tone
        dlyby100msOC(5);
        delay = loCnt; // switch to delay count for 440-Hz tone
    }
}

// ----- ↙
// OC1A interrupt service routine. It starts the next compare operation and clears the OCF1A flag.
// ----- ↙
ISR(TIMER1_COMPA_vect)
{
    OCR1A += delay; // start a new output compare operation
}

// ----- ↙
// The next function creates a time delay that is a multiple of 100 ms using OC1A function.
// ----- ↙
void dlyby100msOC (int cx)
{
    TCCR3A = 0; // configure Timer 1 to operate in normal mode
    TCCR3B = 0x03; // using clkI/0 / 64 as its clock source
    while(cx){
        OCR3A = TCNT3 + 12500; // start Timer 1 OC1A compare operation
        TIFR3 = 1 << OCF3A; // clear the OCF3A flag
        while(!(TIFR3 & (1<<OCF3A))); // wait until OCF3A flag is set
        cx--;
    }
}
```