

```
__C_task void main (void) {
    // инициализация таймера 1
    Tmr1Stop;
    CLRBIT(PRR, PRTIM1); // включить модуль таймера 1
    SETBIT(PLLCSR, LSM); // PLL = 32MHz
    SETBIT(PLLCSR, PLLE); Delay(200*us); do{ } while (!CHKBIT(PLLCSR, PLOCK)); // запуск
PLL
    SETBIT(PLLCSR, PKCE); // разрешить тактирование таймера от PLL
    TCCR1A = BIN(00110011);
    TCCR1B = BIN(01000000); do{ } while (CHKBIT(TCCR1B, PSR1)); // сброс прескалера
    TCCR1C = BIN(00110000);
    TCCR1D = BIN(00000000);
    TCCR1E = BIN(00000000);
    DT1 = 0x00;
    Reg10write(TCNT1reg, 0x000);
    Reg10write(OCR1Areg, 0x3FF);
    Reg10write(OCR1Breg, 0x3FF);
    Reg10write(OCR1Creg, 0x3FF);
    Reg10write(OCR1Dreg, 0x3FF);
    SETBITS(TIFR, BIT(OCF1A)+BIT(OCF1B)+BIT(OCF1D));
    SETBITS(TIMSK, BIT(OCIE1A)+BIT(OCIE1B)+BIT(TOIE1));
    Tmr1Start;
Work:
    // инициализация ADC
    ADMUX = BIN(10011001);
    ADCSRA = BIN(10010110);
    ADCSRB = BIN(00000000);
    ADC = 0;
    SETBIT(DIDR0, AREFD);
    SETBITS(DIDR1, BIT(ADC8D)+BIT(ADC9D));
    // запуск преобразования
    SETBIT(ADCSRA, ADSC); do{ } while (CHKBIT(ADCSRA, ADSC)); ADCdata = ADC;
    // запись результата
    if(ADCdata < ADCmin) ADCdata = ADCmin; // мин. время Toff ключа (обычно 20%)
    PreIntState = __save_interrupt();
    __disable_interrupt();
    AS4value = ADCdata;
    __restore_interrupt(PreIntState);

    goto Work;
}

// Timer/Counter1 Compare Match B
#pragma vector = TIM1_COMPB_vect
__interrupt void TIM1_COMPB(void) {
    unsigned int AStmp = AS4value;
    TC1H = ((AStmp >> 8) & 0x0003);
    OCR1B = (AStmp & 0x00FF); // загрузка величины сравнения для след. цикла
}
```