```
#define F_CPU 1000000U
#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>

#define ped 20
#define ped2 60
int interr=0;
int i = 1;
int j = 1;    //for ADC
int k = 1;    //ISR of timer, k=1 for timer, k-0 for PWM
int w = 1;    //for alarm
int h=1;

int wrong_counter=0;
int timer_is_over=0;

ISR(PORTF_PORT_vect)
{
       int intflags = PORTF.INTFLAGS;
       PORTF.INTFLAGS = intflags;    //breakpoint       //I press step into for 
SW6, SW5, SW5, SW6
       while(i == 1)
       {
              if(PORTF.INTFLAGS == 64)
              {
                     if(PORTF.INTFLAGS == 32)
                     {
                            if(PORTF.INTFLAGS == 32)
                            {
                                   if(PORTF.INTFLAGS == 64)
                                   {
                                          i=0;
                                          timer_is_over = 0;
                                          w=0;   //flag to stop the alarm
                                   }else wrong_counter ++;
                            }else wrong_counter ++;

                     }else wrong_counter ++;
              }else wrong_counter ++;

              if(wrong_counter>2 && j ==0){
                     i = 0;
              }
       }

       if(h==1){
              interr = 1;                //breakpoint
              h++;
       }

       i = 1;
       if(j==1){
              wrong_counter = 0;
       }

}

//ISR TIMER
ISR(TCA0_CMP0_vect)
{
       if(k==1){
        TCA0.SINGLE.CTRLA = 0;              //Disable
    //breakpoint
        int intflags = TCA0.SINGLE.INTFLAGS;
        TCA0.SINGLE.INTFLAGS = intflags;
        interr=1;
        timer_is_over=1;
    }

    if(k==0){
        int intflags = TCA0.SINGLE.INTFLAGS;
        TCA0.SINGLE.INTFLAGS |= intflags;
        PORTD.OUT |= PIN0_bm;       //breakpoint
    }
}

//ISR ADC
ISR(ADC0_WCOMP_vect)
{
    int intflags = ADC0.INTFLAGS;
    ADC0.INTFLAGS = intflags;
    PORTD_OUTCLR = PIN0_bm;
    interr = 1;
    timer_is_over=0;
    j = 0; //breakpoint
}

//ISR PWM
ISR(TCA0_OVF_vect){
    int intflags = TCA0.SINGLE.INTFLAGS;
    TCA0.SINGLE.INTFLAGS = intflags;
    PORTD.OUTCLR |= PIN0_bm;    //breakpoint
}

void timer()
{
    TCA0.SINGLE.CNT = 0;
    TCA0.SINGLE.CTRLB = 0;
    TCA0.SINGLE.CMP0 = ped;
        //breakpoint
    TCA0.SINGLE.CTRLA = TCA_SINGLE_CLKSEL_DIV1024_gc;
    TCA0.SINGLE.CTRLA |= 1;
    TCA0.SINGLE.INTCTRL = TCA_SINGLE_CMP0_bm;
    sei();
    while(interr==0)
    {

    }

    interr = 0;
}

void timer1(){
    TCA0.SINGLE.CNT = 0;
    TCA0.SINGLE.CTRLB = 0;
    TCA0.SINGLE.CMP0 = ped2;
    TCA0.SINGLE.CTRLA = TCA_SINGLE_CLKSEL_DIV1024_gc;
    TCA0.SINGLE.CTRLA |= 1;
    TCA0.SINGLE.INTCTRL = TCA_SINGLE_CMP0_bm;
    sei(); //breakpoint
    while(interr==0)
    {
       }
}

void ADC()
{
       ADC0.CTRLA |= ADC_RESSEL_10BIT_gc;
       ADC0.CTRLA |=ADC_FREERUN_bm;
       ADC0.CTRLA |= ADC_ENABLE_bm;
       ADC0.MUXPOS |= ADC_MUXPOS_AIN7_gc;
       ADC0.DBGCTRL |= ADC_DBGRUN_bm;
       ADC0.WINLT |= 10;
       ADC0.INTCTRL |= ADC_WCMP_bm;                     //breakpoint
       ADC0.CTRLE |= ADC_WINCM0_bm;
       sei();
       ADC0.COMMAND |= ADC_STCONV_bm;
}

void pwm(){
       w=1;
       TCA0.SINGLE.CTRLA = TCA_SINGLE_CLKSEL_DIV1024_gc;
       TCA0.SINGLE.PER = 254;
       TCA0.SINGLE.CMP1 = 127;
       TCA0.SINGLE.CTRLB |= TCA_SINGLE_WGMODE_SINGLESLOPE_gc;
       TCA0.SINGLE.INTCTRL = TCA_SINGLE_OVF_bm;
       TCA0.SINGLE.INTCTRL |= TCA_SINGLE_CMP0_bm;
       TCA0.SINGLE.CTRLA |= TCA_SINGLE_ENABLE_bm;
       sei();
       while(w==1){}
}

int main(void)
{
       PORTD.DIR |= PIN0_bm;

       PORTF.PIN5CTRL |= PORT_PULLUPEN_bm | PORT_ISC_BOTHEDGES_gc;
       sei();

       while(1)
       {
              j = 1;

              while(interr==0)            //breakpoint , SW5=1 to get into ISR
              {

              }
              interr = 0;

              timer();                    //breakpoint

              ADC();                             //breakpoint
              while(interr==0)            //breakpoint
              {

              }
              interr=0;
              timer1();     //breakpoint

              //alarm
              if(timer_is_over == 1 && wrong_counter > 2){
                     k=0;
                     PORTF.PIN5CTRL |= PORT_PULLUPEN_bm |
PORT_ISC_BOTHEDGES_gc;
```

```
            sei();
            pwm(); //breakpoint

        }

        interr = 0;
        h=1;
    }
}
```