

```

/*****
/** WIND PROJECT
****
Controller: ATmega32 (Crystal: 1.000Mhz)
Compiler: WINAVR
Author: Wiertzema und Schroeder, Hamburg, Germany
Date: Jan 2010
****
#include <avr/io.h>
#include <avr/interrupt.h>
#include <util\delay.h>

#ifndef F_CPU
#warning "F_CPU war noch nicht definiert, wird nun nachgeholt mit 4000000"
#define F_CPU 1000000UL // Systemtakt in Hz - Definition als unsigned long beachten
#endif

volatile unsigned long takte;
volatile unsigned long taktebackup;

//Variablen für die Zeit
volatile unsigned int millisekunden=0;
volatile unsigned int sekunde=0;
volatile unsigned int minute=0;
volatile unsigned int stunde=0;

int main(void)
{
    //Timer 0 konfigurieren

    TCCR0 =(1<<WGM01) |(1<<CS01);
    OCR0=125-1; // ### Bugfix 20091221

    //Compare Interrupt aktivieren
    TIMSK|=(1<<OCIE0);

    // Enable INT0 External Interrupt
    GICR |= 1<<INT0;

    // Falling-Edge Triggered INT0 an PD2!
    MCUCR |= 1<<ISC01;

// Enable Interrupts
sei();

DDRA = 0x00; // Alle Pins an A als Eingang
PORTA = 0xff; // Interne Pullups an A ein
DDRD = 0x00; // Alle Pins an D als Eingang
PORTD = 0xff; // Interne Pullups an D ein
DDRB = 0xff; // Alle Pins an B als Ausgang
PORTB = 0xff; // Alle LEDs an B aus

while (1)
{
    if (sekunde == 10)
    {
        sekunde = 0;
        if (takte > 10000) // zu viel Wind
        {
            while (PINA & (1<<PA0)) // bis Bremse angezogen
            {
                PORTB &= ~(1<<PB0); // PB0: Bremsstrom vor (ein)
            }
            PORTB |= (1<<PB0); // PB0: Bremsstrom vor (aus)
            PORTB &= ~(1<<PB2); // PB2: Bremsstrom ist angezogen
            PORTB |= (1<<PB3); // PB3: Erregerstrom (aus)
        }
    }
}
}

```

```

else if (takte > 1000 & takte < 10000) // Wind ok
{
    while (PINA & (1<<PA1)) // bis Bremse geloest
    {
        PORTB &= ~(1<<PB1); // PB1: Bremse rueck (aus)
    }
    PORTB |= (1<<PB1); // PB1: Bremsstrom rueck (aus)
    PORTB |= (1<<PB2); // PB2: Breme ist geloest
    PORTB &= ~(1<<PB3); // PB3: Erregerstrom (ein)
}
else // zu wenig Wind
{
    while (PINA & (1<<PA1)) // bis Bremse geloest
    {
        PORTB &= ~(1<<PB1); // PB1: Bremse rueck (aus)
    }

    PORTB |= (1<<PB1); // PB1: Bremsstrom rueck (aus)
    PORTB |= (1<<PB2); // PB2: Breme ist geloest
    PORTB |= (1<<PB3); // PB3: Erregerstrom (aus)
}
takte = 0;
}
}

// External Interrupt 0 ISR
ISR(INT0_vect)
{
    takte++;
}

// Der Compare Interrupt Handler
// Wird aufgerufen wenn TCNT0 = 125
ISR (TIMER0_COMP_vect)
{
    millisekunden++;
    if(millisekunden==1000)
    {
        sekunde++;
        millisekunden=0;
        if(sekunde==60)
        {
            minute++;
            sekunde=0;
        }
        if(minute ==60)
        {
            stunde++;
            minute=0;
        }
    }
}
}

```