.include "m8def.inc"

.cseg

.org 0

;-------------------------------------------------------------------------------

; Reset and Interrupt vector ;VNr. Beschreibung

;-------------------------------------------------------------------------------

.org 0x000

 rjmp main

.org INT0addr ; External Interrupt0 Vector Address

 reti

.org INT1addr ; External Interrupt1 Vector Address

 reti

.org OC2addr ; Output Compare2 Interrupt Vector Address

 reti

.org OVF2addr ; Overflow2 Interrupt Vector Address

 reti

.org ICP1addr ; Input Capture1 Interrupt Vector Address

 reti

.org OC1Aaddr ; Output Compare1A Interrupt Vector Address

 rjmp TIMER

.org OC1Baddr ; Output Compare1B Interrupt Vector Address

 reti

.org OVF1addr ; Overflow1 Interrupt Vector Address

 reti

.org OVF0addr ; Overflow0 Interrupt Vector Address

 reti

.org SPIaddr ; SPI Interrupt Vector Address

 reti

.org URXCaddr ; USART Receive Complete Interrupt Vector Address

 reti

.org UDREaddr ; USART Data Reg. Empty Interr. Vector Address

 reti

.org UTXCaddr ; USART Transmit Complete Interrupt Vector Address

 reti

.org ADCCaddr ; ADC Interrupt Vector Address

 reti

.org ERDYaddr ; EEPROM Interrupt Vector Address

 reti

.org ACIaddr ; Analog Comparator Interrupt Vector Address

 reti

.org TWIaddr ; Irq. vector address for Two-Wire Interface

 reti

.org INT\_VECTORS\_SIZE

.equ XTAL = 3686400;H

;-------------------------------------------------------------------------------

; Start, Power ON, Reset

;-------------------------------------------------------------------------------

main: ldi r16, LOW(RAMEND) ; Stackpointer initialisieren

 out SPL, r16

 ldi r16, HIGH(RAMEND)

 out SPH, r16

 ldi r16, high( 7200 - 1 )

 out OCR1AH, r16

 ldi r16, low( 7200 - 1 )

 out OCR1AL, r16

 ldi r16, ( 1 << WGM12 ) | ( 1 << CS12 )

 out TCCR1B, r16

 ldi r16, 1 << OCIE1A ; OCIE1A: Interrupt bei Timer Compare

 out TIMSK, r16

 sei

 ldi r16,0

 sts B100,r16

 sts B001,r16

 sts B002,r16

 sts B003,r16

 cbi DDRD,2 ;Eingang "Quittierung"

 sbi PORTD,2 ;Pull Up

 cbi DDRD,3 ;Eingang "Auslösen"

 sbi PORTD,3 ;Pull Up

 cbi DDRD,4 ;Eingang "Überwachung"

 sbi PORTD,4 ;Pull Up

 cbi DDRD,5 ;Eingang "Extern-Error"

 sbi PORTD,5 ;Pull Up

 cbi DDRD,6 ;Eingang "Extern-Ein/Aus"

 sbi PORTD,6 ;Pull Up

 sbi DDRB,0 ;Ausgang "Steuerspannung"

 sbi DDRB,1 ;Ausgang "Zeitmessungs-LED"

 sbi DDRB,2 ;Ausgang "Betriebsbereit"

 sbi DDRB,3 ;Ausgang "Störungs LED"

 sbi DDRB,4 ;Ausgang "Taster-LED"

 sbi DDRB,5 ;Ausgang " Test "

 cbi PORTB,0 ;schalte "Steuerspannung" ein

;-------------------------------------------------------------------------------

mainloop:

 rcall Zeitmessung

 rcall Output

 rcall Futterpause

 rjmp mainloop

;-------------------------------------------------------------------------------

Zeitmessung:lds r16,B100

 lds r17,B001

 lds r18,B002

 sbis PIND,4 ; Messeingang wenn 1,

 rjmp A\_03 ; dann spring zu A\_03

 sbis PIND,2 ; Quittiertaste wenn betätigt wird,

 andi r17,0b11111110 ; dann lösche Time out bit

 sbis PIND,2 ; Quittiertaste wenn betätigt wird,

 ldi r18,0 ; dann lösche Timerregister

A\_03: sbic PIND,4 ; Messeingang wenn nicht 1,

 ldi r18,0 ; dann setze Timer(Zähler) zurück

 sbrc r17,0 ; Wenn Time out bit gesetzt,

 rjmp A\_01 ; dann spring zu A\_01

 sbis PIND,3 ; Auslösetaste auf Patine wenn betätigt,

 ori r17,0b00000001 ; dann setze Time out bit

 sbis PIND,3

 andi r17,0b11110111

 sbic PIND,4 ; Überwachungseingang wenn 0,

 rjmp A\_01 ; dann spring zu A\_01

 sbrs r16,0 ; Timer bit 0 (0,5sec Takt) 0,

 rjmp A\_01 ; dann sping zu A\_01

 sbrc r17,1 ; Flankenauswertung für Timer

 rjmp A\_01

 ori r17,0b00000010

 inc r18 ; Timer wird je sec um 1 erhöht

A\_01: sbrs r16,0 ; Flankenauswertung für Timer

 andi r17,0b11111101

 cpi r18,240 ; Zähler wird ausgewertet (240sec.)

 brne A\_02 ; Wenn nicht 240, spirng zu A\_02

 ori r17,0b00000001

 andi r17,0b11110111

 ldi r18,0

A\_02: sts B001,r17

 sts B002,r18

 ret

;-------------------------------------------------------------------------------

Output: lds r16,B100

 lds r17,B001

 sbrc r17,1 ; Zeitmessungs LED

 sbi PORTB,1 ; "

 sbrs r17,1 ; "

 cbi PORTB,1 ; "

 sbrc r17,0

 rjmp B\_01

 sbi PORTB,2

 sbrs r17,3

 cbi PORTB,0

 rjmp B\_02

B\_01: cbi PORTB,2

 sbi PORTB,0

B\_02: sbis PIND,2

 cbi PORTB,3

 sbrs r17,0

 rjmp B\_03

 sbrs r16,0

 cbi PORTB,3

 sbrc r16,0

 sbi PORTB,3

B\_03: sbrc r17,0

 cbi PORTB,4

 sbrc r17,0

 rjmp B\_04

 sbrs r17,3

 rjmp B\_04

 sbrs r16,0

 cbi PORTB,4

 sbrc r16,0

 sbi PORTB,4

B\_04:

 sts B001,r17

 ret

;-------------------------------------------------------------------------------

Futterpause:lds r16,B100

 lds r17,B001

 lds r18,B003

 sbic PIND,6

 rjmp C\_01

 sbrc r17,0

 andi r17,0b11110111

 sbrc r17,0

 rjmp C\_01

 rcall wait\_1ms

 sbrc r17,2

 rjmp C\_01

 ori r17,0b00000100

 sbrs r17,3

 rjmp C\_02

 andi r17,0b11110111

 cbi PORTB,3

 cbi PORTB,4

 rjmp C\_01

C\_02: ori r17,0b00001000

 sbi PORTB,0

 sbi PORTB,3

C\_01: sbis PIND,6

 rjmp C\_03

 sbrs r17,2

 rjmp C\_03

 rcall wait\_1ms

 andi r17,0b11111011

C\_03: sbrs r17,3

 rjmp C\_04

 sbrs r16,2

 rjmp C\_04

 sbrc r17,4

 rjmp C\_04

 ori r17,0b00010000

 inc r18

C\_04: sbrs r16,2

 andi r17,0b11101111

C\_05: cpi r18,150

 brne C\_06

 ldi r18,0

 andi r17,0b11110111

 cbi PORTB,3

 cbi PORTB,4

C\_06:

 sts B001,r17

 sts B003,r18

 ret

;-------------------------------------------------------------------------------

wait\_1ms: push r16

 push r17

wait\_1ms\_1: inc r16

 brne wait\_1ms\_1

 inc r17

 cpi r17,5

 brne wait\_1ms\_1

 ldi r17,0

 pop r17

 pop r16

 ret

;-------------------------------------------------------------------------------

TIMER: push r16

 lds r16,B100

 inc r16

 sts B100,r16

 pop r16

 reti

;-------------------------------------------------------------------------------

.dseg

B100: .byte 1 ; wird im Interrupt behandelt

B001: .byte 1 ; Statusregister 1

 ; Bit 0: time out Messung oder Test von Taster

 ; Bit 1: Zeitmessungsflanke und LED Zeitmessung

 ; Bit 2: Flankenauswertung Taster Futterpause

 ; Bit 3: Futterpause und Störungs LED ein bei 1

 ; Bit 4: Flankenauswertung Timer 10min.

 ; Bit 5:

 ; Bit 6:

 ; Bit 7:

B002: .byte 1 ; Timerregister 60sec.

B003: .byte 1 ; Timerregister 10min.