

Lösung:

1a)

```
Timer0_ISR:      ; 6
  inc Zeit1      ; 1
  cpi Zeit1, 100; 1
  brlo weiter   ; 1/2
  clr Zeit1      ; 1
  inc Zeit2      ; 1
weiter:
  ldi itmp, VORST; 1
  out TCNT0, itmp; 1
  reti          ; 4
```

1b)

Zeit1 ≤ 99: 6+1+1+2+1+1+100 = 112 μs

Zeit1 > 99: 6+1+1+1+1+1+1+100 = 113 μs

1c)

Statusregister wurde nicht gerettet, Vorstellen nicht am Anfang.

1d)

```
.def isreg = R3 ; SREG retten in ISR
.equ VORST = 256-100+6+2 ; 164

  ldi tmp, 1<<TOIE0
  out TIMSK,tmp ; Overflowinterrupt0 aktiv

Timer0_ISR:      ; 6
  ldi itmp, VORST; 1
  out TCNT0, itmp; 1
  in isreg,SREG ; 1
  inc Zeit1      ; 1
  cpi Zeit1, 100 ; 1
  brlo weiter   ; 1/2
  ldi Zeit1,0    ; 1
  inc Zeit2      ; 1
weiter:
  out SREG,isreg ; 1
  reti          ; 4
```

Zeit: 6+1+1+92 = 100 μs

2)

```
  ldi tmp, 1<<ISC01
  out MCUCR,tmp ; INTO fallende Flanke
  ldi tmp, 1<<INT0
  out GIMSK,tmp ; INTO Interrupt aktiv

INT0_ISR:
  in itmp,TCCR0B
  sbrs itmp,0 ; laeuft timer
  rjmp timerstarten
  ldi itmp,0
  out TCCR0B,itmp; Timer stopp
  cbi PORTB,0 ; PB0 <- 0
  reti

timerstarten:
  ldi Zeit1,VORST
  out TCNT0,Zeit1
  ldi Zeit1,0
  ldi Zeit2,0
  ldi itmp, 1<<CS00
  out TCCR0B,itmp ; Timer Start mit Systemtakt
  reti
```

3)

```
.include "tn2313def.inc"
.def isreg = R3 ; SREG in ISR retten
.def Zeit1 = R16
.def Zeit2 = R17
.def tmp = R18
.def itemp = R19 ; tmp in ISR
.equ VORST = 256-100+6+2 ; 164

rjmp init ; Nach Reset
rjmp INTO_ISR ; ext. Interrupt0
.org OVf0addr
rjmp Timer0_ISR ; Overflow-Interrupt

init:
ldi tmp,low(RAMEND)
out SPL,tmp ; Stack initialisieren
sbi DDRB,0 ; PB0 als Ausgang
ldi tmp, 1<<TOIE0
out TIMSK,tmp ; Overflowinterrupt0 aktiv
ldi tmp, 1<<ISC01
out MCUCR,tmp ; INT0 fallende Flanke
ldi tmp, 1<<INT0
out GIMSK,tmp ; INT0 Interrupt aktiv
sei ; globale Interruptfreigabe
main:
cpi Zeit2,50 ; Zeit2 >= 50
brsh setPB0
rjmp main
setPB0:
sbi PORTB,0 ; PB0 <- 1
rjmp main
```

4a)

$\sim 0x0000 = 0xffff$; $-0x0000 = 0x0000$

$\sim 0xaffe = 0x5001$; $-0xaffe = 0x5002$

4b)

```
.def wartezeit = R20
w200: ; 3 Takte fuer rcall
ldi wartezeit,64 ; 1
wloop:
dec wartezeit ; 1
brne wloop ; 1/2
nop ; 1
ret ; 4
```

Berechnungen:

$200-7 = 193$ Takte sind in dem UP noch zu warten. Die wloop braucht $3 \cdot$ Startwert von wartezeit.

Somit Startwert wartezeit = $193 / 3 = 64$. $193 \% 3 = 1$, ein NOP muss noch eingebaut werden.

4c)