

***** StBySchleife *****

NulldurchgangStBy:

```
cbr Flagregister, pFlanke
rjmp FlankeStBy0
FlankeStBy1:
sbr Flagregister, pFlanke
FlankeStBy0:
sbis pinb, 0
rjmp FlankeStBy1
sbrs Flagregister, 5
rjmp FlankeStBy0
```

StBy:

```
inc Timertaktzaehler
cpi Timertaktzaehler, 50 ; entspricht 1 Sec
brne Sprungziel11
clr Timertaktzaehler
add StBySec, IntSec ; hier die Addition: am Simulator klappts, aber an der Hardware
;läuft der Controller über die 35 Sec, und zwar der Zeit
;entsprechend (Faktor x 5Sec), wie oft ein Interrupt am
;Komparator passiert!!!
;-> d.h. er addiert die im Interrupt verstrichene Zeit im Register
;anscheinend nicht dazu
;BITTE UM HILFE falls Ideen vorhanden

clr IntSec
inc StBySec
cpi StBySec, 35 ;Bei voller Minutenzahl auf 60 stellen.
brne Sprungziel11
clr StBySec
```

rcall Schalter_abschalten

Sprungziel11:
rjmp NulldurchgangStBy

***** Comparator Interruptroutine *****

Comparator:

```
inc ImpulsZaehler
cpi ImpulsZaehler, 2
brne ersterInt
dec ImpulsZaehler
rjmp Timer_Countdown
```

```
ersterInt:  
reti
```

Nulldurchgang:

```
cbr Flagregister, pFlanke  
rjmp Flanke0  
Flanke1:  
sbr Flagregister, pFlanke  
Flanke0:  
sbis pinb, 0  
rjmp Flanke1  
sbrs Flagregister, 5  
rjmp Flanke0
```

Timer_Countdown:

```
sbr Flagregister, Timerroutine  
inc Timertaktzaehler  
cpi Timertaktzaehler, 50  
brne Sprungziel1  
clr Timertaktzaehler  
inc BruehSec  
mov IntSec, BruehSec
```

```
; IntSec wird immer überschrieben, bis 5Sec erreicht  
; sind-> dann reti  
;-> diese Zeit wird dann in der StBy-Schleife addiert,  
; damit die Interrupt-Zeit im ganzen Programm  
; mitberücksichtigt wird  
; geht der µC in reti, so zählt der µC nur Nettozeit der  
; StBy-Schleife, da die Zeit, die im Interrupt verstreicht  
; im Register draufaddiert wird-> siehe StBy-Schleife
```

```
cpi BruehSec, 5
```

```
;Zeit, wo Druckmatic abschaltet
```

```
brne Sprungziel1  
clr BruehSec
```

```
;Bei voller Minutenzahl auf 60 stellen
```

```
reti  
Sprungziel1:  
rjmp Nulldurchgang
```