

AVRASM ver. 1.30 STOP_K_7.ASM Thu Jul 15 02:39:25 2004

```

;***** AT90S1200 *****
;** INFRAPUNASE UNIVERSAALSE TV PULDIGA (LAUDMANN) **
;** STOPPERI ja KELLA JUHTIMIN **
;** ** **
;** Fail:"Stopp_k_n.asm" kus n on versiooni number **
;** Pult : Laudmann programmeeritud koodiga 159 **
;** Kuupäev : 1.7.2004 **
;** Autor : Jüri Puhang **
;** Eeskuju : AVR410.asm **
;** Protsesor : AT90S1200 **
;** ** **
;** SELGITUS **
;** Pult tuleb programmeeride: SET + VCR1 (korraga) ja kui **
;** punane tuli põleb, valida 159 **
;** Programm loeb puldilt koodi ja teeb vastavalt vajutatud **
;** nupule **
;** Kogu kood on 14 bitine **
;** St,St,Ctrl,S4,S3,S2,S1,S0,C5,C4,C3,C2,C1,C0 **
;** Start; Toggle bit; System; Command **
;** ;-----; **
;** ; Kui on kontrollkäsud sees, siis ; **
;** ; PORTD7=Toggle; PORTD6=Start; ; **
;** ; PORTD(5-0) on Command(C5-C0) ; **
;** ;-----; **
;** ** **
;** Protsessori taktsagedus on kvartsiga 8 MHz **
;** NB! Kui sisemine RC generaator on sisse lülitatud, **
;** (Internal RC oscillator enable) siis 1MHz **
;** KELLA ja STOPPERI indikatsiooniks on valgusdiodid **
;** ja neid juhitakse 7- segmendilise dešifraatoriga КР514ИД1 **
;** (ID_vene täded) **
;** Dešifraator 0 kuni 9 1010 - 1011 - **
;** 1100 - 1101 - **
;** 1110 - 1111 - (tühi) **
;** Registriteks, kus hoitakse inditseeritavat infot on **
;** 74LS164N (555ИР8 - IR vene tähed) **
;** ** **
;*****
.include "1200def.inc" ; Seal on AT90S1200 määrangud

;*****
;* APPLICATION NOTE FOR THE AVR FAMILY
;*
;* Number :AVR000
;* File Name : "1200def.inc"
;* Title :Register/Bit Definitions for the AT90S1200
;* Date :99.01.28
;* Version :1.30
;* Support telephone :+47 72 88 43 88 (ATMEL Norway)
;* Support fax :+47 72 88 43 99 (ATMEL Norway)
;* Support E-Mail :avr@atmel.com
;* Target MCU :AT90S1200
;*
;* DESCRIPTION
;* When including this file in the assembly program file, all I/O register
;* names and I/O register bit names appearing in the data book can be used.
;*

```

```

;* The Register names are represented by their hexadecimal addresses.
;*
;* The Register Bit names are represented by their bit number (0-7).
;*
;* Please observe the difference in using the bit names with instructions
;* such as "sbr"/"cbr" (set/clear bit in register) and "sbrs"/"sbrc"
;* (skip if bit in register set/cleared). The following example illustrates
;* this:
;*
;* in   r16,PORTB           ;read PORTB latch
;* sbr  r16,(1<<PB6)+(1<<PB5) ;set PB6 and PB5 (use masks, not bit#)
;* out  PORTB,r16          ;output to PORTB
;*
;* in   r16,TIFR           ;read the Timer Interrupt Flag Register
;* sbrc r16,TOV0           ;test the overflow flag (use bit#)
;* rjmp TOV0_is_set       ;jump if set
;* ...                    ;otherwise do something else
;
;*****
;***** Specify Device
.device AT90S1200

;***** I/O Register Definitions
.equ  SREG  = $3f
.equ  GIMSK = $3b
.equ  TIMSK = $39
.equ  TIFR  = $38
.equ  MCUCR = $35
.equ  TCCR0 = $33
.equ  TCNT0 = $32
.equ  WDTCR = $21
.equ  EEAR  = $1e
.equ  EEDR  = $1d
.equ  EECR  = $1c
.equ  PORTB = $18
.equ  DDRB  = $17
.equ  PINB  = $16
.equ  PORTD = $12
.equ  DDRD  = $11
.equ  PIND  = $10
.equ  ACSR  = $08

;***** Bit Definitions

.equ  INT0   = 6

.equ  TOIE0  = 1

.equ  TOV0   = 1

.equ  SE     = 5
.equ  SM     = 4
.equ  ISC01  = 1
.equ  ISC00  = 0

.equ  CS02   = 2
.equ  CS01   = 1
.equ  CS00   = 0

.equ  WDE    = 3

```

Stopperi programm

```
.equ WDP2 =2
.equ WDP1 =1
.equ WDP0 =0

.equ EEWE =1
.equ EERE =0

.equ PB7 =7
.equ PB6 =6
.equ PB5 =5
.equ PB4 =4
.equ PB3 =3
.equ PB2 =2
.equ PB1 =1
.equ PB0 =0

.equ DDB7 =7
.equ DDB6 =6
.equ DDB5 =5
.equ DDB4 =4
.equ DDB3 =3
.equ DDB2 =2
.equ DDB1 =1
.equ DDB0 =0

.equ PINB7 =7
.equ PINB6 =6
.equ PINB5 =5
.equ PINB4 =4
.equ PINB3 =3
.equ PINB2 =2
.equ PINB1 =1
.equ PINB0 =0

.equ PD6 =6
.equ PD5 =5
.equ PD4 =4
.equ PD3 =3
.equ PD2 =2
.equ PD1 =1
.equ PD0 =0

.equ DDD6 =6
.equ DDD5 =5
.equ DDD4 =4
.equ DDD3 =3
.equ DDD2 =2
.equ DDD1 =1
.equ DDD0 =0

.equ PIND6 =6
.equ PIND5 =5
.equ PIND4 =4
.equ PIND3 =3
.equ PIND2 =2
.equ PIND1 =1
.equ PIND0 =0

.equ ACD =7
.equ ACO =5
.equ ACI =4
```

```

.equ  ACIE  =3
.equ  ACIS1 =1
.equ  ACIS0 =0

.equ  XRAMEND  =0
.equ  E2END    =3F
.equ  FLASHEND =1FF

.equ  INT0addr  =\$001 ;External Interrupt0 Vector Address
.equ  OVf0addr  =\$002 ;Overflow0 Interrupt Vector Address
.equ  ACIaddr    =\$003 ;Analog Comparator Interrupt Vector Address

.def   ZL      =r30
;
;.equ  INPUT    =2   ; PD2 kui IR on PORTD2 küljes
;.equ  INPUT    =4   ; PD4 kui IR on PORTD4 küljes
.equ  INPUT    =6   ; PD6 IR stopperi korral
;
;*****
;
.def   S        =R0   ; Oleku reg hoidmiseks
;         ; katkestuse korral
;
.def   kolmveerand =R1 ; 3/4 IR biti pikkust
.def   viisveerand =R2 ; 5/4 IR biti pikkust
;
.def   sis_koht   =R3 ; Numbripuppudel sisestatud info
;         ; sis_koht on 8 kuni 13
;         ; (seega koht kuhu sisestatakse)
;         ; KELLALE STOPPERILE KOHT
.def   sis_tund   =R4 ; tunnid tund + 10 minutit (8,9)
.def   sis_min    =R5 ; minutid minut + 10 sek (10,11)
.def   sis_sek    =R6 ; sekundid sekund + 1/10 sek(12,13)
;.def  sis_temp   =R7 ; Tööregister sisestamisel

;***** Indikaatoritele väljastatav info
.def   tund       =R8 ; Kaks esimest
.def   min        =R9 ; Kaks keskmist
.def   sek        =R10 ; Kaks viimast
.def   bitiloend  =R11 ; Loendab sisestatud bitte
.def   command    =R12 ; Käsk: Viimased bitid
;         ; S1,S0,C5,C4,C3,C2,C1,C0
.def   system     =R13 ; Ctrl,S4,S3,S2,

.def   stopperi_kask =R14 ; Kasutatakse uue käsu (vajutuse)
;         ; leidmiseka
.def   RESERV      =R15 ;
.def   temp        =R16 ; Tööregister
.def   kymme_ms    =R17 ; Stopperi ja kella 10 millisekund
.def   sada_ms_stop=R18 ; 0,1 sek stopperi pulss
.def   yks_sek_kell =R19 ; 1 sek kella pulss
.def   timerL      =R20 ; IR-i lühike interval
.def   timerH      =R21 ; IR-i pikk interval
;
;
.def   loendi     =R22 ;
;
.def   kontroll    =R23 ; Iga bitt kintrollib kella ja stopperit
; B7 B6 B5 B4="1" - KELL KÄIB

```

Stopperi programm

Leht 5 / Lehti 18

```
;      |      |      + - ="1" - Kellaaja sättimine
;      |      + -
;      + = "1" - On KELLA indikatssioon
;
;      B3      B2      B1      B0="1" - STOPPER KÄIB
;      |      |      + = "0" SOPPERI vaheaeg (indikatsioon seisab)
;      |      + -
;      + = "1" - On STOPPERI indikatsioon
;
;***** Kella registrid
.def   kell_tund   =R24 ; 10h ja 1h
.def   kell_min   =R25 ; 10m ja 1m
.def   kell_sek   =R26 ; 10s ja 1s
;
;***** Stopperi registrid
.def   stop_tund  =R27 ; 1h ja 10m
.def   stop_min   =R28 ; 1m ja 10s
.def   stop_sek   =R29 ; 1s ja 1/10s
.def   ZL         =R30 ; Z-register kaudseks adresseerimiseks
;
.def   timer_temp =R31 ;
;
```

```

.cseg
.org 0
000000 c05a          rjmp  ALGUS

;***** TAIMERI INTERVALID *****
;** TAIMER töötleb katkestust **
;** ** **
;** Peale taimeri programmeerimist ja katkestuste lubamist, **
;** katkestus iga 100 uS tagant ja **
;** IR-ile on - timerL loendab 0,1 mS intervalisid **
;** - timerH loendab 25,6 mS intervalisid **
;** KELL-ale ja STOPPER-ile on **
;** - kymme_ms loendab 10 mS-i ja on pulsiks kellale ning **
;** stopperile **
;** - sada_ms_stop loendab 0,1 sek stopperile ja nullitakse **
;** stopprei startimisel **
;** - yke_sek_kell loendab 1 sek kellale ja töötab pidevalt **
;** **
;*****
.org OVf0addr
000002 b60f  TIMER:
                in      S,SREG          ; Oleku reg. säilitamine
                ;-----; Taimeri loendi alglaadimine
                ;          ; Kui RC gen. on 1 MHz ja CK ei ole
                ;ldi  timer_temp,0xC0   ; jagatud kaheksaga (ei ole CK/8) on
                ;out  TCNT0,timer_temp   ; pulss 64 uS. IR viivised on vaja
                ;-----; muuta.
                ;          ; Kui RC gen. on 8 MHz
000003 e9fc      ldi  timer_temp,0x9C   ; Saadakse 100 uS pulss,
000004 bff2      out  TCNT0,timer_temp   ; ja seda kasutatakse kella, stopperi
                ;-----; ning IR intervalide jaoks

000005 9543      inc  timerL            ; Iga 100 uS +1
                ;          ; Loendab kuni 256x100 uS
                ;          ;
                ;          ;
000006 f409      brne TIMER_KELL_STOP   ; Lisab timerH-ile +1
000007 9553      inc  timerH            ;
                ;          ; Iga 25,6 mS +1

;*****
;** Iga 1/10 sek tagant STOPPERILE + 1 **
;** Iga 1 sek tagant KELLALE + 1 **
;** ** **
;*****

TIMER_KELL_STOP:
000008 9513      inc  kymme_ms          ; kymm_ms loendab kuni 100 x 0,1 ms
                ;          ; = 10 millisekundit ja on stopperi
000009 3614      cpi  kymme_ms,100     ; ja kella pulss
00000a f194      brlt  TIMER_E          ;
00000b 2711      clr  kymme_ms          ;
                ;***** Kas stopper töötab ?
00000c ff70      sbrs  kontroll,0       ; Kui Stopper STOP B0="0"
00000d c00a      rjmp  TIMER_K          ; siis EI loenda
00000e 9523      inc  sada_ms_stop      ; sada_ms_stop loendab 10 x 10 ms
                ;          ; = 100 millisekundit

```

Stopperi programm

Leht 7 / Lehti 18

```

00000f 302a      cpi   sada_ms_stop,10    ; Kui 10 siis lisatakse stopperil
000010 f03c      brlt  TIMER_K            ; 0,1 sek
000011 2722      clr   sada_ms_stop      ; 100 ms uuesti algusesse

;***** Stopperi sekunditele +1 iga 1/10 sek järel
000012 6170      sbr   kontroll,0x10     ; kontroll.B4="1" mis tekitab ind.
                                ; muutumise iga 0,1 sek tagant
000013 95d3      inc   stop_sek          ; 1/10 Sekundi lisamine ja
000014 e0f6      ldi   timer_temp,6     ; kümnendsüsteemi
                                ; korrigeerimine.
000015 0fdf      add   stop_sek,timer_temp
000016 f015      brhs  TIMER_S_S1       ; KUI +6 tekitab tetraadi üle-
000017 50d6      subi  stop_sek,6       ; täitumise, siis
000018 c01b      TIMER_K:
                                rjmp  TIMER_KELL      ; oli tetr. 10 ja seega
                                ; ülekantse järgmise tetraadi
                                TIMER_S_S1:
000019 e6f0      ldi   timer_temp,0x60
00001a 0fdf      add   stop_sek,timer_temp
00001b f010      brcs  TIMER_S_S2       ; 10-süsteemi sekundite
00001c 56d0      subi  stop_sek,0x60    ; korrigeerimine.
00001d c016      rjmp  TIMER_KELL      ; KUI +6 tekitab ületäitumise
                                ; oli 10 sek täis

                                TIMER_S_S2:
00001e 27dd      clr   stop_sek        ;
                                ;***** Stopperi minutitele +1 iga 10 sek järel
00001f 95c3      inc   stop_min         ; Minutite lisamine ja
000020 e0fa      ldi   timer_temp,10    ; 60-süsteemi
                                ; korrigeerimine.
000021 0fcf      add   stop_min,timer_temp
000022 f015      brhs  TIMER_S_M1       ; KUI +6 tekitab tetraadi üle-
000023 50ca      subi  stop_min,10     ; täitumise, siis
000024 c00f      rjmp  TIMER_KELL      ; oli tetr. 6 ja seega
                                ; ülekantse järgmise tetraadi
                                TIMER_S_M1:
000025 e6f0      ldi   timer_temp,0x60
000026 0fcf      add   stop_min,timer_temp
000027 f010      brcs  TIMER_S_M2       ; 60-süsteemi sekundite
000028 56c0      subi  stop_min,0x60    ; korrigeerimine.
000029 c00a      rjmp  TIMER_KELL      ; KUI +10 tekitab ületäitumine
                                ; oli 60 sek täis
                                TIMER_S_M2:
00002a 27cc      clr   stop_min        ;
                                ;***** Stopperi tunnile +1 iga 10 min järel
00002b 95b3      inc   stop_tund        ; Tunnile lisatakse ja
00002c e0fa      ldi   timer_temp,10    ; 60-süsteemi
                                ; korrigeerimine.
00002d 0fbf      add   stop_tund,timer_temp
00002e f015      brhs  TIMER_S_T1       ; KUI +10 tekitab tetraadi üle-
00002f 509a      subi  kell_min,10     ; täitumine, siis
000030 c003      rjmp  TIMER_KELL      ; oli tetr. 6 ja seega
                                ; ülekantse järgmise tetraadi
                                TIMER_S_T1:
000031 3ab0      cpi   stop_tund,0xA0   ;
000032 f409      brne  TIMER_KELL      ; 10-tundi täis
000033 27bb      clr   stop_tund        ;

```

```

;***** Kella sekunditele +1
TIMER_KELL:
000034 9533      inc     yks_sek_kell      ; sada_ms_stop loendab 100 x 10 ms
                                ; = 1 sekund
000035 3634      cpi     yks_sek_kell,100 ; Kui 100 siis lisatakse stopperil
000036 f114      brlt   TIMER_END        ;
000037 2733      clr     yks_sek_kell      ; 1 sekund uuesti algusesse
000038 95a3      inc     kell_sek         ; Sekundi lisamine ja
000039 e0f6      ldi     timer_temp,6     ; kumnendsusteemi
                                ; korrektsoon.

00003a 0faf      add     kell_sek,timer_temp
00003b f015      brhs   TIMER_S1        ; KUI +6 tekitab tetraadi üle-
00003c 50a6      subi   kell_sek,6       ; täitumine, siis
00003d c01b      TIMER_E:
                                rjmp   TIMER_END        ; oli tetr. 10 ja seega
                                ; ülekanne järgmisse tetraadi

TIMER_S1:
00003e eaf0      ldi     timer_temp,0xA0
00003f 0faf      add     kell_sek,timer_temp
000040 f010      brcs   TIMER_S2        ; 60-susteemi sekundite
000041 5aa0      subi   kell_sek,0xA0    ; korrektsoon.
000042 c016      rjmp   TIMER_END        ; KUI +10 tekitab ületäitumise
                                ; oli 60 sek tais

TIMER_S2:
000043 27aa      clr     kell_sek         ;
;***** Kella minutitele +1
000044 9593      inc     kell_min         ; Minutile lisamine ja
000045 e0f6      ldi     timer_temp,6     ; kumnendsusteemi
                                ; korrektsoon.

000046 0f9f      add     kell_min,timer_temp
000047 f015      brhs   TIMER_M1        ; KUI +6 tekitab tetraadi üle-
000048 5096      subi   kell_min,6       ; täitumise, siis
000049 c00f      rjmp   TIMER_END        ; oli tetr. 10 ja seega
                                ; ülekanne järgmisse tetraadi

TIMER_M1:
00004a eaf0      ldi     timer_temp,0xA0
00004b 0f9f      add     kell_min,timer_temp
00004c f010      brcs   TIMER_M2        ; 60-susteemi sekundite
00004d 5a90      subi   kell_min,0xA0    ; korrektsoon.
00004e c00a      rjmp   TIMER_END        ; KUI +10 tekitab ületäitumine
                                ; oli 60 sek täis

TIMER_M2:
00004f 2799      clr     kell_min         ;
;***** Kella tundidele +1
000050 9583      inc     kell_tund        ; Tunnile lisatakse ja
000051 e0f6      ldi     timer_temp,6     ; kumnendsusteemi
                                ; korrektsoon.

000052 0f8f      add     kell_tund,timer_temp
000053 f015      brhs   TIMER_T1        ; KUI +6 tekitab tetraadi üle-
000054 5096      subi   kell_min,6       ; täitumine, siis
000055 c003      rjmp   TIMER_END        ; oli tetr. 10 ja seega
                                ; ülekanne järgmisse tetraadi

TIMER_T1:
000056 3284      cpi     kell_tund,0x24    ;
000057 f409      brne   TIMER_END        ; 24-tundi täis
000058 2788      clr     kell_tund        ;

TIMER_END:
000059 be0f      out    SREG,S           ; Oleku reg. taastamine
00005a 9518      reti                                ; ja katkestuse lõpp

```


Stopperi programm

```

;***** INITSIALISEERIMINE *****
;** Initsialiseeritakse taimer, pordid ja katkestus **
;** Kustutada kella ja stopperi registrid **
;** **
;*****

```

```

00005b e002  ALGUS:  ldi    temp,2      ; Taimer ajapulss/8 CK/8
                ;ldi    temp,1      ; Taimeri loendaja ja ajapulss CK
00005c bf03      out    TCCR0,temp      ;
00005d e9fc      ldi    timer_temp,0x9C    ; Saadakse 100 nS pulss
00005e bff2      out    TCNT0,timer_temp  ;
                ;
00005f e002      ldi    temp,1<<TOIE0     ; Lubada katkestus taimeri
000060 bf09      out    TIMSK,temp      ; ületäitumisel
                ;
                ; PORTD on sisend
000061 e30f      ldi    temp,0b00111111   ; IR-loendi on PD6
000062 bb01      out    DDRD,temp      ; Teised on väljundid
                ;
                ; PORTB on väljund
000063 e30f      ldi    temp,0b00111111   ; TEST ja START
000064 bb07      out    DDRB,temp      ; nupud on sisend
                ; Teised on väljundid
                ;
000065 24cc      clr    command          ; Käsu nullimine
000066 24dd      clr    system          ;
000067 24ee      clr    stopperi_kask   ;
                ;***** Kellaregistrite ja stopperiregistrite nullimine
000068 2466      clr    sis_sek          ; 1/10s ja 1s
000069 2455      clr    sis_min          ; 10s ja 1m
00006a 2444      clr    sis_tund          ; 10m ja 1h
00006b 24aa      clr    sek              ; 1/10s ja 1s
00006c 2499      clr    min              ; 10s ja 1m
00006d 2488      clr    tund              ; 10m ja 1h
00006e 27aa      clr    kell_sek          ; 1 sek ja 10 sek
00006f 2799      clr    kell_min          ; 1 min ja 10 min
000070 2788      clr    kell_tund          ; 1 tund ja 10 tundi
                ; Stopperi ja kella loendid
000071 2711      clr    kymme_ms          ; 10 ms interval
000072 2722      clr    sada_ms_stop       ; 0,1 sek stopperi pulss
000073 2733      clr    yks_sek_kell       ; 1 sek kella pulss
                ;***** Väljundregistrite kustutus
000074 e10f      ldi    temp,0b00011111   ; Kustutada kõik
000075 bb08      out    PORTB,temp      ; reg. 74LS164
000076 0000      nop                    ;
000077 0000      nop                    ;
000078 9ac5      sbi    PORTB, 5      ;
                ;
000079 e978      ldi    kontroll,0x98      ; Sropper seisab ja Kell käib
00007a 9478      sei                    ; Katkestuste lubamine

```

```

;***** PÕHIPROGRAMM *****
;
;** Programm on lõputus tsüklis ja kontrollib **
;** IR-ilt tulevat nivood (PORTD2) **
;** Kui avastatakse start siis: **
;** 1. fikseeritakse LED6-l (kontrolliks) **
;** 2. Mõõdetakse pulsi pikkust ja pannakse paika **
;** 3/4 biti pikkus ja 5/4 biti pikkus **
;** 3. Väetakse vastu 12 bitti mis lähevad **
;** system = Ctrl,S4,S3,S2 **
;** command = S1,S0,C5,C4,C3,C2,C1,C0 **
;** 4. Tehakse stopperi_käsk = Ctrl, C5,C4,C3,C2,C1,C0 **
;** Kui Ctrl muutus, ole võitnud vastu uus nupuvajutus **
;** ja sooritatakse toiming **
;
;*****

```

```

00007b 2755 LOOP: clr timerH ; Alustada 131 mS intervali
; loendamist.

00007c 2744 DET_0: clr timerL ; Alustada lühikeste intervalide
; loendamist.
; cpi timetH,8
00007d 3055 DET_1: cpi timerH,5 ; kas 131 mS on läbi
00007e f008 brlo DET35 ; Kui ei siis kontrollide 3,5 mS-i
; rjmp LOOP ; Kui ja siis uuesti
00007f c09b DET_OUT:
rjmp KELL_STOP_OUT ; Valjastada kellaageg
; cpi timerL,55
000080 3243 DET35: cpi timerL,35 ; Kui 3,5mS madal sisend, siis start
000081 f41c brge START1 ; 55x64 us = 3,52 ms
; 56x64 us = 3,58 ms
000082 9b86 sbis PIND,INPUT ; Esimene Start bit
000083 cff8 rjmp DET_0 ; Madal, siis 3,5 mS viivist uuesti
000084 cff8 rjmp DET_1 ;

; cpi timerH,8
000085 3055 START1: cpi timerH,5 ; Oli start, kuid liiga pikk
; brge LOOP ; üle 130 mS ja otsast peale
000086 f7c4 brge DET_OUT ;
000087 9986 sbic PIND,INPUT ; Ootab starti: Ldks "0"
000088 cffc rjmp START1 ; Startbiti laskuv front
000089 2744 clr timerL ;

; cpi timerL,17
00008a 304b START2: cpi timerL,11 ; Stardi pikkuse mxxtmine
; brge LOOP ; Kui üle 1,1 mS siis otsast peale
00008b f79c brge DET_OUT ; 17x64 uS = 1,09 mS
; 16x64 uS = 1,02 ms
00008c 9b86 sbis PIND,INPUT ; Läks "1"
00008d cffc rjmp START2 ; Startbiti txusev front
00008e 2f04 mov temp,timerL ; 1/2 biti pikkus
00008f 2744 clr timerL ;

000090 2e10 mov kolmveerand,temp ; 1/2 biti pikkus
000091 9416 lsr kolmveerand ; 1/4 biti pikkus
000092 2c21 mov viisveerand,kolmveerand

```

Stopperi programm

Leht 11 / Lehti 18

```

000093 0e10      add    kolmveerand,temp    ; 1/2 + 1/4 = 3/4 biti pikkus
000094 0f00      lsl    temp                ; 1 biti pikkus
000095 0e20      add    viisveerand,temp   ; 1 + 1/4 = 5/4 biti pikkus

000096 1541 START3:cp    timerL,kolmveerand
                                ; Kui St2 > 3/4 biti pikkust
                                ; Siis otsast peale
                                ;brge LOOP
000097 f73c      brge  DET_OUT             ;
000098 9986      sbic  PIND,INPUT         ; St2 lqheb "0"
000099 cffc      rjmp  START3
                                ; St2 laskuv front ja vxib
                                ; alustada

00009a 2744      clr   timerL
00009b 98c6      cbi   PORTB,6            ; Kontroll: Bit6 - Start

00009c e00c      ldi   temp,12
00009d 2eb0      mov   bitiloend,temp     ; Peale starte vxetakse vastu
                                ; 12 bitti
00009e 24cc      clr   command           ; Kustutada kdsk
00009f 24dd      clr   system

                                ;***** BITTIDE VASTUVXTT *****
0000a0 1541 BIT: cp    timerL,kolmveerand
0000a1 f3f0      brlo  BIT                ; Sisestuse hetke ootus

0000a2 9986      sbic  PIND,INPUT         ; Biti lugemine
0000a3 c009      rjmp  BIT_1             ; Kas "1" vqi "0"

0000a4 9488 BIT_0:clc
0000a5 1ccc      rol   command           ; Oli "0" siis C=0
0000a6 1cdd      rol   system            ; Nihutus lqbi C

                                ; Sqnkroniseerimine
0000a7 1542 BIT_0a:cp   timerL,viisveerand
0000a8 f694      brge  LOOP              ; Aeg lqbi, uuesti
0000a9 9b86      sbis  PIND,INPUT         ; Ootab tqisvat fronti
0000aa cffc      rjmp  BIT_0a           ;

0000ab 2744      clr   timerL            ; Taimer otsast
0000ac c008      rjmp  JARG_BIT         ; ja jrgmine bit

0000ad 9408 BIT_1:sec
0000ae 1ccc      rol   command           ; Oli "1" siis C=1
0000af 1cdd      rol   system            ; Nihutus lqbi C

                                ; Sqnkroniseerimine
0000b0 1542 BIT_1a:cp   timerL,viisveerand
0000b1 f64c      brge  LOOP              ; Aeg lqbi, uuesti
0000b2 9986      sbic  PIND,INPUT         ; Ootab laskuvat fronti
0000b3 cffc      rjmp  BIT_1a           ;

0000b4 2744      clr   timerL

                                JARG_BIT:
0000b5 94ba      dec   bitiloend         ; Loendab 12 bitti
0000b6 f749      brne  BIT                ;

```

```

;***** SISESTATUD INFO ANALÜÜS *****
;
; ** Sisestatud on **
; ** system /command **
; ** 3 2 1 0 / 7 6 5 4 3 2 1 0 **
; ** Ctrl, S4, S3, S2, S1, S0, C5, C4, C3, C2, C1, C0 **
; **
;-----; KÄSKUDE KONTROLLIMISEKS:
;
; ; mis on uurimiseks
;ldi temp,0b11111111 ; Kõik on väljundid
;out DDRB,temp ; (ka sisendnupud)
;
;bst system,3 ; Toggle bit läheb
;bld command,7 ; LED7-le (Ctrl)ja vilgub
; ; Uurib sisestatud
;ser temp ; käsu koodi
;eor command,temp ; Inverteerida command
;ldi loendur,0x40
;or command,loendur ; Start vilgub
;out PORTB,command
;rjmp LOOP
;-----;
;*****
0000b7 fad3 bst system,3 ; Ctrl bit-i kontroll,
0000b8 f907 bld temp,7 ; kas on uus kask
;
0000b9 250e eor temp,stopperi_kask
; Kas Ctrl bit muutus
0000ba 7800 andi temp,0x80 ; Kui ole sama käsu kordus,
0000bb f409 brne STOPPER
;rjmp LOOP ; ei anta Stopperile käsku.
0000bc c05e rjmp KELL_STOP_OUT
;***** Stopperi käsu moodustamine *****
STOPPER:
0000bd 2cec mov stopperi_kask,command
; kask on S0,C5,C4,C3,C2,C1,C0
0000be fad3 bst system,3 ; Ctlr bit lisada käsule
0000bf f8e7 bld stopperi_kask,7
;
;-----; KÄSUD KONTROLLIMISEKS
;ser temp
;eor temp,stopperi_kask
;out PORTB,temp
;rjmp LOOP
;-----;
;***** Stopperi 74S164N juhtimine *****
;* Kui nihkeregistrile 74S164N sisenditele 1 ja 2 anda "1"
;* ja clk sisendisse impulss nihkub "1" edasi
;*
;*
;*****
;***** Kustutuskäsu kontroll *****
0000c0 2d0e mov temp,stopperi_kask
0000c1 730f andi temp,0b00111111 ; Eraldada kood
;***** Spooeri kollaste nuppude analüüs *****
0000c2 3101 cpi temp,0b00010001 ; Kas on KUSTUTUS (kollane vasakule)
0000c3 f429 brne S_START
0000c4 27bb CLEAR:clr stop_tund ; Stopperi seisu kustutus
0000c5 27cc clr stop_min ; (koos sada_ms_stop, mis tagab
0000c6 27dd clr stop_sek ; aja mõõtmise 0,01 sek täpsuse,
0000c7 2722 clr sada_ms_stop ; kuna nukkitakse

```

Stopperi programm

Leht 13 / Lehti 18

```

0000c8 c052 S_OUT:rjmp KELL_STOP_OUT ; loendi, nis loendab 10 mS-i)
;
0000c9 3200 S_START:
        cpi    temp,0b00100000 ; Kas on START (kolane ülesse)
0000ca f411        brne  S_STOP      ; SIIS
0000cb 6073        sbr   kontroll,0x03      ; kontroll,B0="1" - START Stopper
; kontroll,B1="1" - Jätkab vaheaega
0000cc c04e        rjmp  KELL_STOP_OUT ;
0000cd 3201 S_STOP:
        cpi    temp,0b00100001 ; Kas on STOP (kollane alla)
0000ce f411        brne  S_VAHE      ; SIIS
0000cf 7f7e        cbr   kontroll,0x01      ; kontroll,B0="0" - STOP Stopper
0000d0 c04a        rjmp  KELL_STOP_OUT ;
0000d1 3100 S_VAHE:
        cpi    temp,0b00010000 ; Kas on VAHEAEG (kollane paremale)
0000d2 f419        brne  S_POWER      ; SIIS
0000d3 e002        ldi   temp,0x02        ; kontroll,B1 inverteerirakse
0000d4 2770        eor   kontroll,temp    ; Iga vajutus kas võtab või jätkab

0000d5 c045        rjmp  KELL_STOP_OUT ; vaheaega.
;
0000d6 300c S_POWER:
        cpi    temp,0b00001100 ; Kui POWER (punane nupp)
0000d7 f419        brne  S_CH_PC      ; SIIS
0000d8 e808        ldi   temp,0x88      ; kontroll,B3 ja B7 inverteerirakse
0000d9 2770        eor   kontroll,temp    ; Iga vajutus kas lüürib indikat-
0000da c040        rjmp  KELL_STOP_OUT ; siooni sisse või välja.
;
0000db 320d S_CH_PC:
        cpi    temp,0b00101101 ; Kui nupp CH*P/C (PUWER-i all)
0000dc f451        brne  S_KELL_ES    ; SIIS
0000dd e200        ldi   temp,0x20      ; kontroll,B5 inverteeritakse
0000de 2770        eor   kontroll,temp    ; Iga peale nupuvajutust vxib
; muuta jooksvat kellaega, kui
; valida numbreid.
0000df ec0f        ldi   temp,0xCF      ; Sisestamise kohas on "kapsaraud"
0000e0 2e40        mov   sis_tund,temp  ; ja ülejäänud tühjad
0000e1 ef0f        ldi   temp,0xFF      ;
;
0000e2 2e50        mov   sis_min,temp   ;
0000e3 2e60        mov   sis_sek,temp   ;
0000e4 e008        ldi   temp,8         ; Sisestada saab Max 6 kohta
0000e5 2e30        mov   sis_koht,temp  ; numbreid ja HOHT on 8 kuni 13
0000e6 c034        rjmp  KELL_STOP_OUT ;
0000e7 300b S_KELL_ES:
        cpi    temp,0b00001011 ; Kellaaja muutmiseks (valitud
0000e8 f411        brne  S_KELL_TE    ; arvuga kella startimine)
; on vaja järjest teha kaks
; nupuvajutuse: Pr ja Nool paremale
0000e9 6470        sbr   kontroll,0x40 ; konrtoll,B6 = "1" on vajutetud Pr
0000ea c030        rjmp  KELL_STOP_OUT ;
;
0000eb 310e S_KELL_TE:
        cpi    temp,0b00011110 ; Kui peale Pr-i on ka nool paremale
0000ec f451        brne  S_NUMBER      ; Siis muudetakse kell dra.
0000ed ff76        sbrs  kontroll,6     ; Kontroll,B6 = "1" siis muuta kell dra
0000ee c02c        rjmp  KELL_STOP_OUT ;
;***** Kellaaja muutmine
0000ef 2d84        mov   kell_tund, sis_tund
0000f0 2d95        mov   kell_min, sis_min ; Kella muutmine
0000f1 2da6        mov   kell_sek, sis_sek ;
0000f2 797f        cbr   kontroll,0x60 ; Kella sättimise ja eelvalike
; Kustutamine

```

Stopperi programm

Leht 14 / Lehti 18

```

0000f3 6970      sbr   kontroll,0x90      ; Kell käima ja indikatsioon
                                ; välja.
0000f4 c026      rjmp  KELL_STOP_OUT    ;
0000f5 c025      rjmp  KELL_STOP_OUT    ;
0000f6 cf84      S_LOOP:
                                rjmp  LOOP ;

;***** Vajutatud puldi nupu analüüs *****
S_NUMBER:
;-----; KONTROLL-KÄSUD
;                               ; Sisestatud info stopperile
;andi temp,0x0F ; 1/10 sek
;mov  sis_tund,temp; 10 sek
;mov  sis_min,temp ; 10 min
;mov  sis_sek,temp ;
;-----; KONTROLL-KÄSUD
;                               ;
;ldi  temp, 0x32 ; Mis arv välja
;mov  sis_tund,temp;
;ldi  temp, 0x54 ;
;mov  sis_sek,temp ;
;ldi  temp, 0x76 ;
;mov  sis_min,temp ;
;-----;
; KUHU                               ;
; 1/10s & 1s ; 10s & 1m ; 10m & 1h ;
; PD3 ; PD4 ; PD5 ;
;-----;
0000f7 3202      cpi   temp,0b00100010   ; Nupp 12 ?
0000f8 f411      brne  S_NUM_1           ; Kui, siie tagurpidi c
0000f9 e00b      ldi   temp,0x0B        ;
0000fa c007      rjmp  S_NUM_2
0000fb 7300      S_NUM_1:
                                andi  temp,0b00110000   ; Kas on numbrinupp
0000fc f4f1      brne  KELL_STOP_OUT    ; Ei olnud numbrid
0000fd 2d0e      mov   temp,stopperi_kask
0000fe 700f      andi  temp,0b00001111   ; Eraldada number uuesti
0000ff 300a      cpi   temp,0x0A        ; Nupp 11 ?
000100 f409      brne  S_NUM_2
000101 e00f      ldi   temp,0x0F        ; 1111 - tühik
000102 2d63      S_NUM_2:
                                mov   loendi,sis_koht   ; Sisestatava info koht
000103 9566      lsr   loendi           ; Z-registrisse.
000104 2fe6      mov   ZL,loendi        ; sis_koht Z-reg
                                ; (8,9) 4=R4
                                ; (10,11) 5=R5
                                ; (12,13) 6=R6
000105 8160      ld    loendi,Z         ; R4,R5 vxi R6-e laadimine
000106 fe30      sbrs  sis_koht,0       ; Kui HOHT on paarisarv,
000107 c00b      rjmp  S_NUM_VASAK     ; Siis number vasakule
;***** Number parempoolsesse dekaadi
000108 7f60      andi  loendi,0xF0      ; Number läheb par. dekaadi
000109 2b60      or    loendi,temp      ; Vasakut detraadi ei muudeta
00010a 8360      st    Z,loendi         ; Tagasi R4,R5 vxi R6
00010b 9433      inc   sis_koht         ;
00010c e00e      ldi   temp,14          ; kui ole ыle 13
00010d 1630      cp    sis_koht,temp     ; See tđhendab:
00010e f051      breq  S_NUM_TAIS       ; kõik on sisestatud
00010f 95e3      inc   ZL               ;
000110 ec0f      ldi   temp,0xCF        ; "kapsaraud" järgmisse

```

```

000111 8300      st      Z,temp
000112 c008      rjmp   KELL_STOP_OUT
                S_NUM_VASAK:
000113 e06c      ldi    loendi,0x0C      ; "kapsaraud" paremp. dekaadi
000114 9502      swap  temp              ; Number läheb vasusse dekaadi
000115 2b60      or     loendi,temp      ; Paremat detraadi ei muudeta
000116 8360      st     Z,loendi         ;
000117 9433      inc   sis_koht          ;
000118 c002      rjmp   KELL_STOP_OUT
                S_NUM_TAIS:      ; Uuesti sisestamise algusesse,
                                ; kuid ilma "kapsarauata" esimesel
000119 e008      ldi    temp,8           ; kohal. Sisestada saab Max 6 kohta
00011a 2e30      mov   sis_koht,temp     ; numbreid ja HOHT on 8 kuni 13
                                ;

```

```

;***** Kell ja stopperi valjastamine indikaaturitele

```

```

KELL_STOP_OUT:

```

```

;-----; KÄSKUDE KONTROLLIMISEKS:  **
;      ; mis on uurimiseks          **
;ldi   temp,0b11111111 ; Kõik on väljundid          **
;out   DDRB,temp       ; (ka sisendnupud)          **
;      ; Näitab KELLA LED-idel      **
;      ;                               **
;ser   temp             ; sekundid, minutid      **
;eor   temp,sek         ; voi tunnid              **
;eor   temp,min         ;                               **
;eor   temp,tund        ;                               **
;out   PORTB,temp      ;                               **
;rjmp  LOOP             ;                               **
;-----;

```

```

;***** STOPPER (all)

```

```

00011b fd73      sbrcc  kontroll,3      ; Kui STOPPER-i indikatsiooni ei ole
00011c c005      rjmp   K_STOPPER_NAIT ; (B3=0) siis indikatsioon
00011d ef0f      ldi    temp,0xff      ; kustutatakse ära.
00011e 2e80      mov   tund,temp       ; 1111 - kood on 7-segmen dilise
00011f 2e90      mov   min,temp        ; depifraatori jaoks tühi
000120 2ea0      mov   sek,temp        ; (kõik kustutatud)
000121 c004      rjmp   K_STOPPER

```

```

;***** Mida näidata STOPPERIL (alla)

```

```

K_STOPPER_NAIT:

```

```

000122 2e8b      mov   tund,stop_tund  ;
000123 2e9c      mov   min,stop_min   ;
000124 2ead      mov   sek,stop_sek   ;
000125 fd71      sbrcc  kontroll,1    ; Kui VAHEAEG, siis indikatsiooni
                                ; ei muudeta (B1="0")

```

```

;-----; KONTROLLIMISEKS
;sbrs  kontroll,5     ; Kas kella sättimine ?
;rjmp  K_STOPPER_KELL ; SIIS
;mov   tund,sis_tund  ; Sisestatav info indi-
;mov   min,sis_min    ; kaatoritele
;mov   sek,sis_sek    ;
;rjmp  K_STOPPER      ;
;-----; KONTROLLIMISEKS
K_STOPPER_KELL:      ; Kui kell on all
;mov   tund,kell_tund ;
;mov   min,kell_min   ;
;mov   sek,kell_sek   ;
;-----;

```

```

K_STOPPER:

```

```

000126 d012      rcall  OUT_STOP
                ;***** KELL (üleval)
000127 fd77      sbrc   kontroll,7      ; Kui KELL-a indikatsiooni ei ole
000128 c006      rjmp   K_KELL_NAIT      ; (B7=0) siis indikatsioon
000129 ef0f      ldi    temp,0xff      ; kustutatakse dra.
00012a 2e80      mov    tund,temp      ; 1111 - kood on 7-segmen dilise
00012b 2e90      mov    min,temp      ; depifraatori jaoks t̄hi
00012c 2ea0      mov    sek,temp      ; (kõik kustutatud)
00012d d03a      rcall  OUT_KELL      ; Väljastada
00012e cf4c      rjmp   LOOP
                ;***** Mida näidata KELLAL (üleval)
                K_KELL_NAIT:
00012f 2e88      mov    tund,kell_tund      ;
000130 2e99      mov    min,kell_min      ;
000131 2eaa      mov    sek,kell_sek      ;
000132 ff75      sbrs   kontroll,5      ; Kas kella sätimine ?
000133 c003      rjmp   K_KELL_OUT      ; SIIS
000134 2c84      mov    tund,sis_tund      ; Sisestatav info indi-
000135 2c95      mov    min,sis_min      ; kaatoritele
000136 2ca6      mov    sek,sis_sek      ;
                K_KELL_OUT:
000137 d030      rcall  OUT_KELL      ; ei muudeta (B4="0")
000138 cf42      rjmp   LOOP      ;

                ;***** Alamprogramm INDIKATSIOON ALLA *****
                ;**          tund,minut,sekund          **
                ;** Kui STOPPER, siis          **
                ;**          tund+10 min / min + 10 sek / 1 sek + 1/10 sek          **
                OUT_STOP:
000139 ff74      sbrs   kontroll,4      ; Kontroll,B4 on iga 0,1 sek
00013a 9508      ret      ; järei "1" ja seega iga kord
                ; ind ie muudeta(muidu vilguks)
00013b e300      ldi    temp,0b00110000      ; Avada stopperi
00013c bb08      out   PORTB,temp      ; regtstri
                ; laadimine

                ;
                ;***** 1/10 sek ja sek registri laadimine *****

00013d e068      ldi    loendi,8      ; Kaheksa korda
                ; nihutada
00013e 2d0a      mov    temp,sek      ;
00013f 9502      swap  temp      ; Vahetada El. skeemi
                YKS_PLS:      ; pärast
000140 1f00      rol   temp      ; Nihutada Carry bitti
000141 f010      brcs  YKS_YKS      ;
000142 98c0      cbi   PORTB,0      ; 1/10 Sek 0 bit
000143 c001      rjmp  YKS_CLOCK      ;
                YKS_YKS:      ;
000144 9ac0      sbi   PORTB,0      ;
                YKS_CLOCK:
000145 9a93      sbi   PORTD,3      ; Clock=1
000146 0000      nop      ;
000147 9893      cbi   PORTD,3      ; Clock=0
000148 956a      dec  loendi
000149 f7b1      brne  YKS_PLS
                ;***** 10 sek ja 1 min registri laadimine *****
00014a e068      ldi    loendi,8      ; Kaheksa korda
                ; nihutada
00014b 2d09      mov    temp,min      ;

```


Stopperi programm

Leht 17 / Lehti 18

```

00014c 9502      swap  temp          ; Vahetada El. skeemi
                KYM_PLS:          ; pärast
00014d 1f00      rol    temp          ; Nihutada Carry bitti
00014e f010      brcs  KYM_YKS
00014f 98c1      cbi   PORTB,1       ; 1/10 Sek 0 bit
000150 c001      rjmp  KYM_CLOCK    ;
                KYM_YKS:          ;
000151 9ac1      sbi   PORTB,1       ;
                KYM_CLOCK:        ;
000152 9a94      sbi   PORTD,4       ; Clock=1
000153 0000      nop
000154 9894      cbi   PORTD,4       ; Clock=0
000155 956a      dec  loendi
000156 f7b1      brne  KYM_PLS
                ;***** 10 min ja tunni registri laadimine *****
000157 e068      ldi   loendi,8      ; Kaheksa korda
                ; nihutada
000158 2d08      mov  temp,tund      ;
000159 9502      swap  temp          ; Vahetada El. skeemi
                MIN_PLS:          ; pärast
00015a 1f00      rol    temp          ; Nihutada Carry bitti
00015b f010      brcs  MIN_YKS
00015c 98c2      cbi   PORTB,2       ; 1/10 Sek 0 bit
00015d c001      rjmp  MIN_CLOCK    ;
                MIN_YKS:          ;
00015e 9ac2      sbi   PORTB,2       ;
                MIN_CLOCK:        ;
00015f 9a95      sbi   PORTD,5       ; Clock=1
000160 0000      nop
000161 9895      cbi   PORTD,5       ; Clock=0
000162 956a      dec  loendi
000163 f7b1      brne  MIN_PLS
000164 e200      ldi   temp,0b00100000 ; Sulgeda stopperi
000165 bb08      out  PORTB,temp    ; regstri
                ; laadimine
000166 7e7f      cbr  kontroll,0x10  ; Kustutada vilkumise
000167 9508      ret    ; bitt kontroll,B4
                ;***** Alamprogramm INDIKATSIOON ÜLESSE *****
                ;**          tund, minut, sekund          **
                ;** (Normaalselt KELL)                    **
                ;** Kui kell siis 10tundi + tund/ minutid/ sekundid **
                OUT_KELL:
000168 ff74      sbrs  kontroll,4    ; Kontroll,B4 on iga 0,1 sek
000169 9508      ret    ; järei "1" ja seega iga kord
                ; ind ie muudeta(muidu vilguks)
00016a e208      ldi   temp,0b00101000 ; Avada kella
00016b bb08      out  PORTB,temp    ; regstri
                ; laadimine

                ;***** 1/10 sek ja sek registri laadimine *****

00016c e068      ldi   loendi,8      ; Kaheksa korda
                ; nihutada
00016d 2d0a      mov  temp,sek       ;
00016e 9502      swap  temp          ; Vahetada El. skeemi
                K_YKS_PLS:        ; pärast
00016f 1f00      rol    temp          ; Nihutada Carry bitti
000170 f010      brcs  K_YKS_YKS
000171 98c0      cbi   PORTB,0       ; 1 Sek 0 bit
000172 c001      rjmp  K_YKS_CLOCK  ;
                K_YKS_YKS:        ;

```

Stopperi programm

Leht 18 / Lehti 18

```

000173 9ac0      sbi    PORTB,0      ;
                K_YKS_CLOCK:
000174 9a90      sbi    PORTD,0      ; Clock=1
000175 0000      nop
000176 9890      cbi    PORTD,0      ; Clock=0
000177 956a      dec    loendi
000178 f7b1      brne   K_YKS_PLS
                ;***** 10 sek ja 1 min registri laadimine *****
000179 e068      ldi    loendi,8     ; Kaheksa korda
                ; nihutada

00017a 2d09      mov    temp,min;
00017b 9502      swap  temp          ; Vahetada El. skeemi
                K_KYM_PLS:
                ; pärast
00017c 1f00      rol    temp          ; Nihutada Carry bitti
00017d f010      brcs   K_KYM_YKS
00017e 98c1      cbi    PORTB,1      ; 1 Min 0 bit
00017f c001      rjmp  K_KYM_CLOCK
                K_KYM_YKS:
                ;
000180 9ac1      sbi    PORTB,1      ;
                K_KYM_CLOCK:
                ;
000181 9a91      sbi    PORTD,1      ; Clock=1
000182 0000      nop
000183 9891      cbi    PORTD,1      ; Clock=0
000184 956a      dec    loendi
000185 f7b1      brne   K_KYM_PLS
                ;***** 10 min ja tunni registri laadimine *****
000186 e068      ldi    loendi,8     ; Kaheksa korda
                ; nihutada

000187 2d08      mov    temp,tund
000188 9502      swap  temp          ; Vahetada El. skeemi
                K_MIN_PLS:
                ; pärast
000189 1f00      rol    temp          ; Nihutada Carry bitti
00018a f010      brcs   K_MIN_YKS
00018b 98c2      cbi    PORTB,2      ; 1 Tund 0 bit
00018c c001      rjmp  K_MIN_CLOCK
                K_MIN_YKS:
                ;
00018d 9ac2      sbi    PORTB,2      ;
                K_MIN_CLOCK:
                ;
00018e 9a92      sbi    PORTD,2      ; Clock=1
00018f 0000      nop
000190 9892      cbi    PORTD,2      ; Clock=0
000191 956a      dec    loendi
000192 f7b1      brne   K_MIN_PLS
000193 e200      ldi    temp,0b00100000 ; Sulgeda kella
000194 bb08      out   PORTB,temp    ; regtstri
                ; laadimine
                ;
000195 7e7f      cbr   kontroll,0x10 ; Kustutada vilkumise
000196 9508      ret    ; bitt kontroll,B4

```

Assembly complete with no errors.