```
;Datei: Sender_31.asm
.include<m8def.inc>
.cseg

/*
R16:    temp1
R17:    temp2
R18:    Empfangendes High-Byte v. SPI-Schnittstelle
R19:    Empfangendes Low-Byte v. SPI-Schnittstelle
R20:    CMD_HighByte
R21:    CMD_LowByte
*/

Init:
    ldi r16,LOW(RAMEND)
    out spl,r16
    ldi r16,HIGH(RAMEND)
    out sph,r16
        ;IO-Init + Pull-Ups
    clr r16
    out ddrc,r16        ;PC zur Eingabe
    ser r16
    out portc,r16       ;Pull-Ups der Eingänge
    out ddrd,r16        ;PD zur Ausgabe
        ;SPI-Init
    ldi r16,0b00101100  ;MOSI=1, SS=1, SCK=1
    out ddrb,r16
    out portb,r16       ;??
    ldi r16,0b01010011  ;SPCR: SPIE(0),SPE(1),DORD(0),MSTR(1),CPOL(0),CPHA(0),SPR1(1),S
PR0(1)
    out spcr,r16
        ;RFM12-Init
    ;0x80d8
    ldi r20,0x80
    ldi r21,0xd8
    rcall WriteCMD
    ;0x8208
    ldi r20,0x82
    ldi r21,0x08
    rcall WriteCMD
    ;0xA640
    ldi r20,0xA6
    ldi r21,0x40
    rcall WriteCMD
    ;0xC647
    ldi r20,0xC6
    ldi r21,0x47
    rcall WriteCMD
    ;0x90A0
    ldi r20,0x94
    ldi r21,0xA0
    rcall WriteCMD
    ;0xC2AC
    ldi r20,0xC2
    ldi r21,0xAC
    rcall WriteCMD
    ;0xCA80
    ldi r20,0xCA
    ldi r21,0x81
    rcall WriteCMD
    ;0xCA83
    ldi r20,0xCA
    ldi r21,0x83
    rcall WriteCMD
    ;0xC49B
    ldi r20,0xC4
    ldi r21,0x9B
    rcall WriteCMD
    ;0x9850
    ldi r20,0x98
    ldi r21,0x50
    rcall WriteCMD
    ;0xE000
```

```
    ldi r20,0xE0
    ldi r21,0x00
    rcall WriteCMD
    ;0xC80E
    ldi r20,0xC8
    ldi r21,0x0E
    rcall WriteCMD
    ;0xC000
    ldi r20,0xC0
    ldi r21,0x00
    rcall WriteCMD

Start:
    ;0x8228          ;Open PA
    ldi r20,0x82
    ldi r21,0x28
    rcall WriteCMD
    NOP
    NOP
    NOP
    NOP
    ;0x8238
    ldi r20,0x82
    ldi r21,0x38
    rcall WriteCMD
    NOP
    NOP
    ;0xB8AA
    ldi r21,0xAA
    rcall WriteFSKbyte
    rcall WriteCMD
    ;0xB8AA
    ldi r21,0xAA
    rcall WriteFSKbyte
    rcall WriteCMD
    ;0xB8AA
    ldi r21,0xAA
    rcall WriteFSKbyte
    rcall WriteCMD
    ;0xB82D
    ldi r21,0x2D
    rcall WriteFSKbyte
    rcall WriteCMD
    ;0xB8D4
    ldi r21,0xD4
    rcall WriteFSKbyte
    rcall WriteCMD
    ;0xB8FF              ;Datenbyte mit Testgröße 55
    ldi r21,0x55
    rcall WriteFSKbyte
    rcall WriteCMD
    ;0xB8AA
    ldi r21,0xAA
    rcall WriteFSKbyte
    rcall WriteCMD
    ;0x8208              ;Close PA
    ldi r20,0x82
    ldi r21,0x08
    rcall WriteCMD
    ;0x8200              ;receive end, enter sleep
    ldi r20,0x82
    ldi r21,0x00
    rcall WriteCMD
    ldi r16,0x01        ;LED=1
    out portd,r16
    rcall zeit_100ms
    ldi r16,0x00        ;LED=0
    out portd,r16
    rcall zeit_1s
    rjmp start


WriteFSKbyte:
```

```
    push r21
    push r20
    ldi r20, 0x00
    ldi r21, 0x00
    rcall WriteCMD
    pop r20
    pop r21
    ldi r20,0xB8
    rcall WriteCMD
    ret

WriteCMD:
    cbi portb,2         ;/SS PB2 ->"low"
    out SPDR,r20
    Warte_Senden:
        sbis SPSR,SPIF  ;warte bis high-Byte übertragen ist
        rjmp Warte_Senden
    in r18,SPDR         ;r18 s.o.
    out SPDR,r21
    Warte_Senden_1:
        sbis SPSR,SPIF  ;warte bis low-Byte übertragen ist
        rjmp Warte_Senden_1
    in r19,SPDR         ;r19 s.o.
    sbi portb,2
    ret



zeit_100ms:
    push r16
    push r17
    push r18
    in r16,sreg
    push r16
    ldi r16,0x05    ;100ms   0x19
zeit_100ms_1:
    ldi r17,0xcd    ;20ms
zeit_100ms_2:
    ldi r18,0x1d    ;100µs
zeit_100ms_3:
    dec r18
    brne zeit_100ms_3
    dec r17
    brne zeit_100ms_2
    dec r16
    brne zeit_100ms_1
    pop r16
    out sreg,r16
    pop r18
    pop r17
    pop r16
    ret

zeit_1s:
    push r16
    push r17
    push r18
    in r16,sreg
    push r16
    ldi r16,0x32    ;1000ms  FA
zeit_1s_1:
    ldi r17,0xcd    ;20ms  c8
zeit_1s_2:
    ldi r18,0x1d    ;100µs  c8
zeit_1s_3:
    dec r18
    brne zeit_100ms_3
    dec r17
    brne zeit_100ms_2
    dec r16
    brne zeit_100ms_1
    pop r16
    out sreg,r16
```

```
    pop r18
    pop r17
    pop r16
    ret
```