```
    .include <m2560def.inc>
    .cseg
    .org    0x00
    rjmp    start
    .org    0xF6
start:  ldi r20,low(RAMEND) ; set up stack pointer
    out SPL,r20     ;   "
    ldi r20,high(RAMEND);   "
    out SPH,r20     ;   "
    ldi r16,3       ; configure USART1 to MSPI mode and shift data
    ldi r17,0       ; at 2 MHz
    call    initMegaMSPI    ;   "
; send out configuration data to all three MC14489s
    ldi r28,7
    cbi PORTD,PD0   ; enable SPI transfer to MC14489
    ldi ZL,low(confDat<<1) ; set up configuration data pointer
    ldi ZH,high(confDat<<1);    "
loop1:  lpm r16,Z+
    call    putch_USART_MSPI
    dec r28
    brne    loop1
    sbi PORTD,PD0
; send out display data to 3 MC14489s
    ldi r28,9       ; set up loop count
    ldi ZL,low(dispDat<<1)  ; set up table pointer
    ldi ZH,high(dispDat<<1) ;   "
    cbi PORTD,PD0   ; enable SPI transfer to MC14489
loop2:  lpm r16,Z+      ; transfer display data to MC14489
    call    putch_USART_MSPI    ;   "
    dec r28     ;   "
    brne    loop2       ;   "
    sbi PORTD,PD0   ; update seven-segment displays
forever:nop
    rjmp    forever

confDat: .db    0x01,0x00,0x00,0x01,0x00,0x00,0x45
dispDat: .db    0x88,0x01,0x12,0x81,0x20,0x00,0xC2,0x67,0xFC
; ---------------------------------------------------------------------------------------
; This subroutine initializes the USART1 of the MEGA AVR to MSPI mode, and also enales
; reception and transmission. The setting of baud rate is passed in r17:r16.
; ---------------------------------------------------------------------------------------
initMegaMSPI:
    clr r20
    sts UBRR0H,r20
    sts UBRR0L,r20
    in  r20,DDRD
    andi    r20,0xD4    ; clear bit 5,3,1,0
    ori r20,0x29    ; set bit 5, 3, and 0
    out     DDRD,r20    ; configure PD5/XCK1, PD3/TXD1 for output
    ldi r20,(1<<RXEN1)|(1<<TXEN1) ; enable receiver and transmitter
    sts UCSR1B,r20
    ldi r20,(1<<UMSEL11)|(1<<UMSEL10)|(0<<UCPHA1)|(0<<UCPOL1)
    sts UCSR1C,r20  ; select MSPI mode
    sts UBRR1H,r17  ; set up baud rate
    sts UBRR1L,r16  ;   "
    ret
; ---------------------------------------------------------------------------------------
; The following subroutine outputs a character in r16 to USART1 in MSPI mode.
; ---------------------------------------------------------------------------------------
putch_USART_MSPI:
```

```
    lds r20,UCSR1A  ; wait until USART data register is empty
    sbrs    r20,UDRE1   ;   "
    rjmp    putch_USART_MSPI ;  "
    sts UDR1,r16    ; start data transmission
waitTXC1:
    lds r20,UCSR1A  ; wait until the character has been shifted out
    sbrs    r20,RXC1    ;   "
    rjmp    waitTXC1    ;   "
    lds r22,UDR1    ; clear RXC1 flag
    ret
```