

```

;29.09.2023 test09
;
; zum testen einer subtractions-routine
;.....
; PIC: 18F14K22//Assembler:MPASM V5.43//IDE: MPLABX v5.20//PICKit3
;
;mclre enabled = normale io-nutzung somit disabled = a3 nur für programming
;
;
#include <p18F14K22.inc>

;Config settings
CONFIG IESO = OFF, PLEN = OFF, FOSC = IRC, FCEN = OFF, PCLKEN = OFF
CONFIG BOREN = OFF, BORV = 19, PWRTEN = OFF, WDTEN = OFF
CONFIG MCLRE = ON, HFOFST = OFF, DEBUG = OFF, STVREN = ON
CONFIG XINST = OFF, BBSIZ = OFF, LVP = OFF
CONFIG CP0 = OFF, CP1 = OFF
CONFIG CPD = OFF, CPB = OFF
CONFIG WRT0 = OFF, WRT1 = OFF
CONFIG WRTB = OFF, WRTC = OFF, WRTD = OFF
CONFIG EBTR0 = OFF, EBTR1 = OFF
CONFIG EBTRB = OFF

errorlevel -302 ;surpress the 'not in bank0' warning

GPR_VAR          UDATA
adclow            RES      1
adchigh           RES      1
eeplow            RES      1
eephigh           RES      1
sub               RES      1
;.....
Org 0x0000          ;kein reset vector
bra START

MAIN_PROG CODE

START:
movlw b'00000010' ;Setup main init
movwf OSCCON       ;set cpu clock speed of 31KHz
clrf OSTUNE        ;move contents of working register into OSCCON
;

clrf LATA
clrf TRISA         ;a3 ist nur als eingang implementiert
clrf ANSEL         ;dig buffer enabled
movlw b'11111111' ;alle pull-ups-r's enabled
movwf WPUA
clrf LATB
clrf TRISB
clrf LATC
clrf TRISC
;
;-----
MainLoop:          ;beginn hauptprogramm
nop
main01:
movlw d'140'       ;für TEST-ROUTINE hier die 8-bit-werte für adc eingeben
movwf adclow
movlw d'00'        ; 2 empfohlene beispiele: erst dez 140, danach dez 160
movwf adchigh
main04:
movlw d'17'        ;eep ist in fast allen fällen größer,
movwf eeplow       ;deshalb eep minus adc
clrf eepphigh
nop

```

```

    call    subtrahier    ;EEP MINUS ADC
    nop
    bnn     main05        ;ergebnis ist zero oder positiv: jetzt anzeigen
    bra     hoeher        ;negativ : adc ist größer als eeprom: stufe höher
;-----
main05:                                ;ergebnis subtract ist positiv
    nop
    bra     MainLoop      ;ein durchlauf fertig
hoeher:                                ;ergebnis subtract ist negativ
    nop
    nop
    bra     main04        ;zum erneuten einpassen
;-----
                                ;eingänge: adc in adclow u adchigh
subtrahier:                          ;          eeprom in eeplow u eepleigh
                                ;eep ist in fast allen fällen größer als adc,
sub1:                                ;deshalb eep minus adc
    movff   eepleigh,sub    ;eepleigh soll nicht überschrieben werden: sichern
    movf    adclow,0,1      ;adclow in WREG
    subwf   eeplow,0,1      ;eep minus adc (WREG),ergebnis in WREG
    bnn     sub2            ;wenn positiv nach sub2
    decf    eepleigh        ;ist negativ: für nächsten subtract negativer
                                ;übertrag in eepleigh
sub2:
    movf    adchigh,0,1     ;adchigh in WREG
    subwf   eepleigh,0,1    ;eeprom minus adc (WREG),ergebnis in WREG
    movff   sub,eepleigh    ;eepleigh zurück sichern

    return
;-----
    end
;-----

```