

;Längenmessung: Aufgabe Nr 8 aus µP-Aufgabensammlung
;(c) Stefan Lehmann

;Deklaration des Prozessortyps
device 16F84

;Deklaration diverser Symbolnamen

indirekt	EQU	0	;Register indirect
pcl	EQU	2	
status	EQU	3	;Statusregister an Adr. 3
fsr	EQU	4	;Zeigerregister für ind. Adressierung
porta	EQU	5	;Adresse 5 auf Bank 0
portb	EQU	6	
trisa	EQU	5	;Adresse 5 auf Bank 1
trisb	EQU	6	

cflag	EQU	0	;Carryflag
zflag	EQU	2	;Zeroflag
rp0	EQU	5	;Bankumschaltbit
signal	EQU	0	;Eingangssignal an RA,0

counter0	EQU	10H	;LSB des 4-stelligen Zählers
counter1	EQU	11H	
counter2	EQU	12H	
counter3	EQU	13H	

ii	EQU	0CH	;Zählvariable
alterPegel	EQU	0DH	;für Flankenerkennung
Digit	EQU	0EH	;Hilfregister Digitanwahl

signalmaske	EQU	00000001B	
digitmaske	EQU	11110001B	;je nach Ansteuerung der Digits

signaleingang	EQU	0	;Bit 0 von Port RA
---------------	-----	---	--------------------

cold:

GOTO	start	;Unterprogramme überspringen
------	-------	------------------------------

;Unterprogramm FLANKE

;UP liest nur den Port ein, maskiert das richtige Bit und macht eine

;XOR-Verknüpfung mit dem alten Pegel

flanke

MOVF	porta,W	;Port RA nach W lesen
------	---------	-----------------------

ANDLW	signalmaske	;nur RA0 ist gültig
XORWF	alterPegel, W	;mit vorherigen Pegel vergleichen
RETURN		;bei ZFlag = 1, kein Pegelwechsel

;Unterprogramm INIT_FSR

;setzt die notwendigen Variablen auf deren Startwert. Das Setzen von
 ;DIGIT ist eigentlich nur bei der Anzeigeroutine notwendig, schadet aber
 ;beim anderen Aufruf nicht.

init_fsr

CLRF	digit	;nur für Anzeigeroutine wichtig
MOVLW	4	;Anzahl der vorhandenen Stellen
MOVWF	ii	;ii = Laufvariable
MOVLW	counter0	;Startadresse in W laden
MOVWF	FSR	;und ins Zeigerregister laen
RETURN		

;DISP gibt den Inhalt von COUNTER auf der 7-Segmentanzeige aus

disp

CALL	init_fsr	
DECF	ii	;niederwert. Stelle nicht anzeigen
INCF	fsr	;deshalb auch Zeiger um eins weiter

disp1

MOVF	indirekt, W	;Wert holen
CALL	convert	;in Bitmuster für 7-Seg. umsetzen
MOVWF	portb	;Segmentinf anlegen
CALL	digitimpuls	
MOVLW	3	;Zeiger in 3er Schritten erhöhen
ADDWF	digit	
INCF	fsr	
DECFSZ	ii	;hier max. 3 x
GOTO	disp1	
RETURN		;danach fertig

;DIGITIMPULS erzeugt je nach anzuzeigener Stelle einen Impuls an RA1,
 RA2 oder RA3

digitimpuls

MOVF	digit, W	;Digitinfo nach W lesen
ADDWF	pcl	;an die entsprech. Position springen
BCF	porta, 1	;impuls an RA,1 erzeugen
BSF	porta, 1	;Impuls wieder egnehmen
RETURN		
BCF	porta, 2	;Impuls an RA,2

BSF	porta,2	
RETURN		
BCF	porta,3	;impuls an RA,3
BSF	porta,3	
RETURN		

;CONVERT wandelt den Wert in W in das passende Bitmuster einer 7-Segmentanzeige um

convert

ANDLW	15	;max. 16 Tabelleinträge
ADDWF	pcl	
RETLW	00111111B	;Muster für die 0
RETLW	00000110B	;für 1
RETLW	01011011B	;für 2
RETLW	01001111B	;3
RETLW	01100110B	
RETLW	01101101B	;5
RETLW	01111101B	;6
RETLW	00000111B	;7
RETLW	01111111B	;8
RETLW	01101111B	;9
RETLW	01110111B	;A
RETLW	01111100B	;b
RETLW	00001111B	;C
RETLW	01011110B	;d
RETLW	01111001B	;E
RETLW	01110001B	;F

start

;Initialisierung der Ports und Variablen

init:

;man darf einer Adresse 2 Namen geben

BSF	status,rp0	;auf Bank 1 umschalten
MOVLW	11110001B	;RA1 bis RA3 Ausgänge für Digit
MOVWF	trisa	;RA0 bleibt Eingang
CLRF	trisb	;Port RB komplett auf Ausgang
BCF	status,rp0	;zurück auf Bank 0
MOVLW	15	;Digitselekt auf 1 setzen
MOVWF	porta	

reset

CLRF	counter0	;Zähler auf 0 stellen
CLRF	counter1	

CLRF	counter2
CLRF	counter3

;Hauptschleife. Hier wird zuerst der Zählerstand angezeigt und
;dann auf die Flanke gewartet

display

CALL	disp
------	------

main

CALL	flanke	;bei ZFlag = 1 ist Flanke da
BTFSC	status, zflag	;
GOTO	main	;keine Flanke, weiter warten

flanke_da

XORWF	alterPegel	;neue Flanke merken
BTFSS	alterPegel,signaleingang	;richtige Flanke?
GOTO	main	;war falsche Flanke

flanke_ok

CALL	init_fsr	;FSR für ind. Adressierung setzen
INCF	indirekt	;niedrigste Stelle um 2 hochzählen

counting

INCF	indirekt	;
MOVF	indirekt,W	;hochgezählter Wert in W
XORLW	10	;ist der Wert 10 erreicht?
BTFSS	status, zflag	
GOTO	display	;zurück und Wert anzeigen

overflow

CLRF	indirekt	;aktuelle Stelle löschen
INCF	fsr	;Zeiger um einen Schritt weiter
DECFSZ	ii	;max. 4 Durchläufe
GOTO	counting	;nächste Stelle bearbeiten
GOTO	display	;zurück und Wert anzeigen

END