
(c) 2007 Gerhard Günzel

Datei: SOLARREGLER.BAS
Datum: 09.03.2007
Letzte Aenderung:

Beschreibung:

Einfacher Solarregler, der Kesseltemperatur und Kollektortemperatur misst, auswertet und eine Umwälzpumpe ein- bzw ausschaltet. Frei einstellbar sind Werte für Anschalttemperatur und Abschalttemperatur der Pumpe sowie Pumpe immer ein. Eingabe über Rotary-Switch.

Ablauf:

Es werden laufend die Temperaturen von Kessel und Kollektor gemessen, angezeigt und ausgewertet. Steigt die Temperatur am Kollektor über die des Kessels (um die eingestellte Temperaturdifferenz für Anschalttemperatur), dann schaltet die Pumpe ein. Ist die Temperatur am Kollektor unterhalb des eingestellten Wertes für Abschalttemperatur, dann schaltet die Pumpe wieder ab. Ist "Pumpe immer eingeschaltet" programmiert, so wird nur die Temperatur gemessen und angezeigt jedoch nicht ausgewertet. D.H. die Pumpe läuft ständig.

Fühler1 (PT 1000) an PortC.0 - Heizkessel
Fühler2 (PT 1000) an PortC.1 - Kollektor

Rotary-Switch zur Dateneingabe schaltet immer GND
Taster leitet Dateneingabe ein und beendet diese
Rotary-Switch rechts drehen Parameter erhöhen bzw. ja
Rotary-Switch links drehen Parameter verkleinern bzw. nein
Rotary-Switch Taster an PD2 = Int.0
Rotary-Switch Kanal A an PD3
Rotary-Switch Kanal B an PD4

Eingabedaten werden im EEPROM permanent gespeichert
Temperaturdifferenz Anschalten Adresse &H10
Temperaturdifferenz Abschalten Adresse &H20
Pumpe immer an Adresse &H30

Anzeige über LCD Display 2 x 24
erste Zeile Kesseltemperatur und Kollektortemperatur
zweite Zeile Delta Abschalttemp. Delta Anschalttemperatur und Pumpenzustand
Im Eingabemenue-Zustand erste Zeile - Klartext zur Eingabe
Im Eingabemenue-Zustand zweite Zeile - Parameter, die über R-Switchdrehung eingegeben werden

Settings:

Compilersettings REGFILE, BAUD, LCD,
Config LCD und Debouncezeit
DIM der Variablen
Declaration von SUBs
Ports festlegen
Interruptroutinen festlegen
ADC konfigurieren
Interrupts konfigurieren

Programmablauf:

Auslesen der Parameter aus dem EEPROM und anzeigen.
DO LOOP Schleife zum Temperaturen messen, anzeigen und schalten.
Drücken des Rotary-Switch löst einen Interrupt aus, der lediglich einen Merker setzt, der dann in der DO LOOP Schleife erkannt wird und eine SUB-Routine zur Dateneingabe einbindet. Danach wird der Merker wieder gelöscht.
In der Routine werden nacheinander alle drei Parameter entweder geändert oder bestätigt und dann im EEPROM gespeichert. Drehen links oder rechts ändert und zeigt Parameter im LCD (2. Zeile) an. Taster vom R-Switch bestätigt, speichert den Wert und geht zur nächsten der drei Eingabeschleifen.
Die drei Routinen sind einfach gehalten.
Zuerst wird im LCD (erste Zeile) angezeigt was man einstellt.
In einer Schleife werden nacheinander über debounce zuerst der Drehschalter und dann der R-Switches abgefragt. Wird der Drehschalter erkannt, wird über eine SUBroutine des debounce Befehls die Drehrichtung erkannt und der Wert incrementiert oder decrementiert. Der Wert des Parameters wird in der 2. Zeile des LCD angezeigt.
Wird der Taster über debounce erkannt, dann wird die Schleife verlassen und der Parameter im EEPROM gespeichert. Dann wird die nächste Schleife abgearbeitet.
Die dritte Schleife zeigt bei Rechtsdrehung "JA" an und bei Linksdrehung "NEIN".
Der Parameter eine BYTE-Variable wird entweder auf 11111111 oder 00000000 gesetzt und im EEPROM gespeichert.
Die DO LOOP Schleife zweigt bei gesetztem Interruptmerker in die Eingabe Routine und löscht den Merker dann wieder. Dann kommt die Messung der Kesseltemperatur.
Hier wird der ADC für der ersten Kanal gestartet.
Eine ADC Interrupt Routine holt zeitverzögert den Wert des Kanals in Variablen, rechnet die Temperatur aus und zeigt diese im LCD an. Dann wird der ADC für den zweiten Kanal genauso abgearbeitet. Dann wird geprüft, ob die Pumpe immer an ist und wenn ja im LCD "PUMP PER" angezeigt; oder die Temperaturdifferenz beider Kanäle ausgewertet und entsprechend die Pumpe an oder abgeschaltet.

```

'
'
'
'----- Standard-Init von BASCOM -----
$regfile = "m8def.dat"
$baud = 9600
$crystal = 8000000
Config Lcdpin = Pin , Db4 = Portb.0 , Db5 = Portb.1 , Db6 = Portc.2 , Db7 = Portc.3 , E = Portd.7 ,
Rs = Portd.6
Config Lcd = 24 * 2
Config Debounce = 2                               '2Msec. für Rotary-Switch
'-----
Dim Temp_kessel As Single
Dim Temp_kollektor As Single
Dim Lowert As Byte
Dim Hiwert As Byte
Dim Temp_diff As Single
Dim Temp_pumpe_aus As Single
Dim Temp_pumpe_an As Single
Dim Merker_interrupt0 As Bit
Dim Merker_pumpe_permanent_an As Byte
Declare Sub Menue                               'Einstellung der Parameter und Pumpe direkt
Declare Sub Count_delta_aus                     'Subroutine für Delta Pumpe aus
Declare Sub Count_delta_an                     'Subroutine für Delta Pumpe an
Declare Sub Pumpe_permanent_an                 'Subroutine Pumpe immer an

Ddrb.2 = 1                                       'PortB.1 als Ausgang f. Relais

'----- Hier Werte für Temperaturdifferenz Pumpe eingeben
Temp_pumpe_aus = 5                              'Differenz zur AB-schaltung
Temp_pumpe_an = 12                             'Differenz zur AN-schaltung

'----- Taster und Rotaryswitch initialisieren
Config Portd.2 = Input                          'Int0 als Eingang - Taster
Portd.2 = 1                                    'Pullup ein
Config Portd.3 = Input                          'Int1 als Eingang R-switch A-Signal
Portd.3 = 1                                    'pullup ein
Config Portd.4 = Input                          'pullup ein - Port D,4 Input - R-switch B-Signal
Portd.4 = 1

'----- Interrupt Routinen festlegen
On Int0 Int0_setup                             'Setuproutine bei gedrücktem R-Switch
On Adc Onadc                                   'ADC-Interrupt Routine

'----- Ext. Interrupts initialisieren
McuCr = &B00001010                            ' Int0 und Int1 falling Edge
Gicr.6 = 1                                     'Int0 ein
Gicr.7 = 0                                     'Int1 aus

Sreg.7 = 1                                     'Interrupt generell frei Statusregister 7

'----- AD-Wandler initialisieren
Ddrc = &B00001100                             'Eingang 0,1 PortC.0.C1 ICPin=23/24
Admux = &B11000000                            'Aref intern, Bit0=adc0 oder adc1
Adcsra = &B10001111                           'Bit7-ADC an
                                                'Bit6-Start
                                                'Bit5-free run
                                                'Bit4-Intflag
                                                'Bit3- Int ena
                                                'Bit2,1,0-Prescale select-128

'----- Hauptprogramm -----
Cls                                             'LCD löschen
'----- EEPROM auslesen und anzeigen
Readeeprom Temp_pumpe_aus , &H10              'Temp.differenz Pume aus
Readeeprom Temp_pumpe_an , &H20              'Temp.Differenz Pume ein
Readeeprom Merker_pumpe_permanent_an , &H30  'Merker Pumpe immer an
Temp_pumpe_aus = Int(temp_pumpe_aus)        'Kommastellen weg
Temp_pumpe_an = Int(temp_pumpe_an)          'Kommastellen weg
'----- Anzeige An-Abschaltung Pumpe und Pumpe immer an
Locate 2 , 1
Lcd Temp_pumpe_aus ; " :"; Temp_pumpe_an
If Merker_pumpe_permanent_an = &HFF Then
Locate 2 , 15
Lcd "Pumpe PER"
End If

Do
'----- Prüfen ob Rotary-Switch gedrückt war

```

```

If Merker_interrupt0 = 1 Then
    Call Menue
    Locate 2 , 1
    Lcd Temp_pumpe_aus ; " :." ; Temp_pumpe_an
    Reset Merker_interrupt0
End If
'Int.0 setzt Merker
'Zum Menue für Parametereingabe
'Parameter anzeigen
'Merker vom Int.0 RESET

'----- Fühler Kessel messen und anzeigen
Admux.0 = 0
Set Adcsra.6
Waitms 20
Temp_kessel = Hiwert * 256
Temp_kessel = Temp_kessel + Lowert
Temp_kessel = Temp_kessel - 468
Temp_kessel = Temp_kessel * 0.541
Temp_kessel = Round(temp_kessel )
Locate 1 , 1
Lcd "Kess " ; Temp_kessel
'----- Fühler Kollektor messen und anzeigen
Admux.0 = 1
Set Adcsra.6
Waitms 20
Temp_kollektor = Hiwert * 256
Temp_kollektor = Temp_kollektor + Lowert
Temp_kollektor = Temp_kollektor - 468
Temp_kollektor = Temp_kollektor * 0.541
Temp_kollektor = Round(temp_kollektor )
Locate 1 , 15
Lcd "Koll " ; Temp_kollektor
'----- An- und Abschaltung der Pumpe - einbezogen Merker für Pumpe permanent an
Temp_diff = Temp_kollektor - Temp_kessel
If Merker_pumpe_permanent_an = &H00 Then
    If Temp_diff => Temp_pumpe_an Then
        Set Portb.2
        Locate 2 , 15
        Lcd "Pumpe an "
    ElseIf Temp_diff =< Temp_pumpe_aus Then
        Reset Portb.2
        Locate 2 , 15
        Lcd "Pumpe aus"
    End If
Elseif Merker_pumpe_permanent_an = &HFF Then
    Locate 2 , 15
    Lcd "Pumpe PER"
    Set Portb.2
End If
Loop

'----- Subroutine zur Dateneingabe über Rotary-Switch
Sub Menue
    Cls
    Locate 1 , 1
'----- Temperaturdifferenz zur Abschaltung der Pumpe setzen
    Lcd "Delta_Temp. Pumpe aus"
Label1_pumpe_aus:
    Debounce Pind.3 , 0 , Count_delta_aus , Sub
    Locate 2 , 1
    Lcd Temp_pumpe_aus
    Debounce Pind.2 , 0 , Label1_pumpe_aus1
Goto Label1_pumpe_aus
Label1_pumpe_aus1:
    Writeeeprom Temp_pumpe_aus , &H10

'----- Temperaturdifferenz zur Anschaltung der Pumpesetzen
    Locate 1 , 1
    Lcd "Delta_Temp. Pumpe an "
Label1_pumpe_an:
    Debounce Pind.3 , 0 , Count_delta_an , Sub
    Locate 2 , 1
    Lcd Temp_pumpe_an
    Debounce Pind.2 , 0 , Label1_pumpe_an1
Goto Label1_pumpe_an
Label1_pumpe_an1:
    Writeeeprom Temp_pumpe_an , &H20

'----- Pumpe permanent ein oder aus
    Locate 1 , 1
    Lcd "Pumpe immer an "
Label1_pumpe_immer_an:
    Debounce Pind.3 , 0 , Pumpe_permanent_an , Sub
    Locate 2 , 1
    'R-Switch drehen
    'R-Switch drücken
    'R-Switch drehen
    'R-Switch drücken
    'R-Switch drehen

```

```

If Merker_pumpe_permanent_an = &HFF Then
  Lcd "ja"
Elseif Merker_pumpe_permanent_an = &H00 Then
  Lcd "nein"
End If
Debounce Pind.2 , 0 , Lapel_pumpe_immer_an1      'R-Switch drücken
Goto Lapel_pumpe_immer_an
Lapel_pumpe_immer_an1:
  Writeeprom Merker_pumpe_permanent_an , &H30

  Cls
End Sub

'----- Subroutinen -----
'----- Subroutine Temp für Abschaltung eingeben
Sub Count_delta aus
  If Pind.4 = 1 Then
    Decr Temp_pumpe_aus
  Elseif Pind.4 = 0 Then
    Incr Temp_pumpe_aus
  End If
End Sub
'----- Subroutine Temp für Anschaltung eingeben
Sub Count_delta an
  If Pind.4 = 1 Then
    Decr Temp_pumpe_an
  Elseif Pind.4 = 0 Then
    Incr Temp_pumpe_an
  End If
End Sub
'----- Subroutine Merker für Pumpe immer ein oder aus setzen
Sub Pumpe_permanent_an
  If Pind.4 = 1 Then
    Merker_pumpe_permanent_an = &B00000000
  Elseif Pind.4 = 0 Then
    Merker_pumpe_permanent_an = &B11111111
  End If
End Sub

'----- Interrupt-Routinen -----
'----- ADC-Interrupt gibt aktuellen Wert des AD-Converter zurück
Onadc:
  Lowert = Adcl      'lesen
  Hiwert = Adch
Return

'----- Ext-Int.0 setzt Merker, der in Hauptprogramm ausgewertet wird
Int0_setup:
  Set Merker_interrupt0
  Waitms 100      'Zeit zur Tastenentprellung R-Switch
Return

End      'end program

```