

```

test-carte8.bas
'-----
'  ----- COMMANDE CIRCULATEUR CHAUFFE-EAU SOLAIRE -----
'  ----- COPYLEFT CHRISTIAN TRILLAUD trillaud@gmail.com -----
'-----

'##### constantes #####
'----- PIC -----
CONST DEVICE = 3B                                'pic utilisé

'----- variables de fonctionnement -----
dim deltaON as byte
dim Tchlimit as byte
dim tempov3 as byte
dim tempoBOUCLE as byte
dim Tchmin as byte
dim Nmax as byte
dim version as byte

'----- initialisation des variables -----
'-----
deltaON = 4                                'différence de température pour enclencher le fonctionnement (en
degrés)
Tchlimit = 90                              'température maxi du haut de cuve (en degrés)
tempov3 = 40                               'durée du démarrage en vitesse 3 (en secondes)
tempoBOUCLE = 50                           'durée de la boucle principale - sans tenir compte des autres
temps (en ms)
Tchmin = 48                                'valeur minimale de Tch pour obtenir ECS à 38°C (en degrés)
alarme LED
Nmax = 50                                  'temporisation de 1 affichage des choix de mode auto / manu, de
mode manu (en 0,1 seconde)
version = 8                                'version du programme

'----- I2C -----
Const SCL = 11                              'port I2C
Const SDA = 12                              'port I2C

'----- DS2482 adresses de canaux d'acquisition -----
Const CH0 = &HF0                            'channel P0 - pin 14
Const CH1 = &HE1                            'channel P1 - pin 15
Const CH2 = &HD2                            'channel P2 - pin 16
Const CH3 = &HC3                            'channel P3 - pin 1

```

Const CH4 = &HB4	test-carte8.bas
Const CH5 = &HA5	'chanel P4 - pin 12
Const CH6 = &H96	'chanel P5 - pin 11
Const CH7 = &H87	'chanel P6 - pin 10
	'chanel P7 - pin 9
'----- SORTIES -----	
CONST SV1 = 14	'port sortie pour commande vitesse 1
CONST SV3 = 15	'port sortie pour commande vitesse 3
'----- LEDS -----	
Const LEDR1 = 5	'led rouge clignotante sur Tch < Tchmin (nécessaire pour ECS)
Const LEDR2 = 6	'led rouge témoin circulateur en vitesse 1 (normal)
Const LEDJ = 7	'led jaune témoin circulateur en vitesse 3 (pour monter colonne
eau)	
Const LEDV = 8	'led verte témoin circulateur en arrêt
'----- Inters -----	
Const bpBleu = 16	'bouton poussoir bleu
Const bpRouge = 17	'bouton poussoir rouge
'##### variables #####	
'----- Variables temperature -----	
'--- mesures dans l'eau	
dim Trc as byte	'température de l'eau en sortie de capteur solaire (le capteur
échange-t-il bien ?)	
dim Tch as byte	'température de l'eau en haut de la cuve
dim Tcb as byte	'température de l'eau en bas de la cuve (stratification ?)
dim Tecs as byte	'température de l'eau chaude sanitaire (le serpentin échange-t-il
bien ?)	
'--- mesures dans l'air	
dim Tcap as byte	'température de l'air dans le capteur solaire
dim Tint as byte	'température de l'air de la maison
dim Text as byte	'température de l'air extérieur
'--- mesures dans l'eau	
dim Src as byte	'signe de la température de l'eau en sortie de capteur solaire
(le capteur échange-t-il bien ?)	
dim Sch as byte	'signe de la température de l'eau en haut de la cuve
dim Scb as byte	'signe de la température de l'eau en bas de la cuve
(stratification ?)	

```

test-carte8.bas

'--- mesures dans l'air
dim Secs as byte
échange-t-il bien ?)
dim Scap as byte
dim Sint as byte
dim Sext as byte

'----- Variables 1-wire -----
Dim CH as byte
Dim KL as byte
Dim KH as byte
Dim DA as byte
Dim BY as byte
Dim Sig as byte
Dim Tf as byte
Dim TL as byte
Dim TH as byte
Dim TT as byte

'----- Variables de fonctionnement -----
dim tempovar as integer
dim j as integer
dim n as byte
dim etatCIRC as byte
dim etatCircTemp as byte
dim Tseuil1 as byte
dim Tseuil2 as byte
dim Tchi as byte
dim autoManu as byte
dim autoManuTemp as byte
dim bouton as byte
dim Tmax as byte

'##### initialisation #####

'curseur off
CSROFF

'variables
autoManu = 1

'signe de la température de l'eau chaude sanitaire (le serpent
'signe de la température de l'air dans le capteur solaire
'signe de la température de l'air de la maison
'signe de la température de l'air extérieur

'Channel lecture : num de l'entrée capteur considérée
'Octet de poids faible DS18B20
'Octet de poids fort DS18B20
'variable de commande du DS18B20
'test busy
'signe
'valeur décimale
'4 bits poids faible
'3 bits poids fort
'temperature

'pour sub TEMP01S
'pour sub TEMP01S
'pour sub affichMODE
'etat commande circulateur
'etat commande circulateur temporaire (mode manu)
'temperature seuil sub auto (calcul)
'temperature seuil sub auto (calcul)
'temperature cuve haute initiale
'variable d etat du type de gestion 1=auto 0=manu
'variable temporaire d autoManu
'variable acquisition des boutons poussoir
'variable pour declencher l'arret

'gestion mode auto par default

```

# test-carte8.bas

```
'led
out LEDR1,0
out LEDR2,0
out LEDJ,0
out LEDV,1

'led pb temperature Tch < Tchmin eteinte
'led circulateur en vitesse3 eteinte
'led circulateur en vitesse1 eteinte
'debut commande led circulateur au repos
```

```
'circulateur non commandé
GOSUB CIRC_STOP
```

```
'##### PROGRAMME PRINCIPAL #####
```

```
'mesure initiale des températures
GOSUB MESURES
```

```
'affichage du choix du mode de gestion
GOSUB affichMODE
```

```
'affichage du fond de l'afficheur (caracteres qui ne changent pas)
GOSUB affichFOND
```

DEBUT:

```
'mesure des températures
GOSUB MESURES
```

```
'affichage des temperatures
GOSUB affichTEMP
```

```
'mode de fonctionnement
if autoManu = 1 then GOSUB AUTO
```

```
'affichage de l'etat de l'eau dans la cuve (utilisable ou non)
GOSUB affichETAT
```

```
delay tempoBOUCLE
```

```
'changement mode ?
bouton = keyin(bpRouge,50)
if bouton = 0 then
  CLS
  delay 1000
```

```
'test bp rouge (changement)
```

```

                                GOSUB affichMODE
                                GOSUB affichFOND
                                end if

GOTO DEBUT

'##### SUBROUTINES #####
AUTO:
  '-- commande du circulateur en mode auto
  if Tch >= Tchlimit then
    if etatCIRC = 1 then
      GOSUB CIRC_STOP                                'arret du circulateur si temperature trop elevee
    else
      return
    end if
  else
    Tseuil2 = Tch + deltaON                            'calcul temperature declenchement
    'commande d arret du circulateur
    if etatCIRC = 1 then
      Tseuil1 = Tmax - 1                                'hysteresis
      if Tch > Tmax then
        Tmax = Tch                                    'memorisation valeur max
      elseif ((Tch < Tseuil1) AND (Tcap < Tseuil2)) then
        GOSUB CIRC_STOP                                'arret ciculateur si temperature cuve en baisse ET temp
      end if
    end if
    'commande de marche du circulateur
    else
      if Tcap >= Tseuil2 then
        GOSUB CIRC_START                                'declenchement circulateur si temp capteur sup bas cuve +
      end if
    end if
  end if
return

'----- affichage du fond d'ecran qui ne change pas (varIN : etatCIRC)-----
affichFOND:

```

# test-carte8.bas

```
CLS
'ligne 1
LOCATE 0,0
if etatCIRC = 1 then
    PRINT "Capt Air/Eau"
else
    print "Air Capteur"
end if

'ligne 2
LOCATE 0,1
PRINT "Haut Cuve"

'ligne 3
LOCATE 0,2
PRINT "Bas Cuve"

'ligne 4
LOCATE 0,3
PRINT "int      ext"

return

'----- affichage des températures (varIN : Trc, Tch, Tcb, Tecs, Tcap, Text, Tint)-----
affichTEMP:

'--- capteur Temp air capteur
LOCATE 13,0
if Tcap < 127 then
    PRINT DEC(Tcap,3,1)
else
    PRINT "---"
end if
'signe
if Tcap > 9 then
    LOCATE 13,0
else
    LOCATE 14,0
end if
if Scap = 1 then
    PRINT "-"
else
    print " "
```

```
end if

'--- capteur eau retour capteur
if etatCIRC = 1 then
    LOCATE 16,0
    print "/"
    if Trc < 127 then
        PRINT DEC(Trc,2,1)
    else
        PRINT "--"
    end if
else
    locate 16,0
    print " "
end if

'--- capteur Temp cuve haute
LOCATE 11,1
if Tch < 127 then
    PRINT DEC(Tch,3,1)
else
    PRINT "---"
end if

'--- capteur Temp cuve haute initiale
if etatCIRC = 1 then
    LOCATE 16,1
    PRINT "( )"
    LOCATE 17,1
    PRINT DEC(Tchi,2,1)
else
    LOCATE 16,1
    PRINT " "
end if

'--- capteur Temp cuve basse
LOCATE 11,2
if Tcb < 127 then
    PRINT DEC(Tcb,3,1)
else
    PRINT "---"
end if
```

test-carte8.bas

```
'--- capteur Temp interieure
LOCATE 4,3
if Tint < 127 then
    PRINT DEC(Tint,2,1)
else
    PRINT "--"
end if

'--- capteur Temp exterieure
LOCATE 12,3
if Text < 127 then
    PRINT DEC(Text,2,1)
else
    PRINT "--"
end if
'signe
if Text > 9 then
    LOCATE 11,3
else
    LOCATE 12,3
end if
if Sext = 1 then
    PRINT "-"
else
    print " "
end if

'---auto/manu
locate 16,3
if autoManu = 1 then
    print "AUTO"
else
    print "MANU"
end if

return

'----- mesure des temperatures -----
MESURES:
    CH = CH0
    GOSUB LECTURE
```



# test-carte8.bas

```

Trc = TT
Src = Sig

CH = CH1
GOSUB LECTURE
Tch = TT
Sch = Sig

CH = CH2
GOSUB LECTURE
Tcb = TT
Scb = Sig

'
'
'
'
CH = CH3
GOSUB LECTURE
Tecs = TT
Secs = Sig

CH = CH4
GOSUB LECTURE
Tcap = TT
Scap = Sig

CH = CH5
GOSUB LECTURE
Tint = TT
Sint = Sig

CH = CH6
GOSUB LECTURE
Text = TT
Sext = Sig

return

'----- LECTURE D'UN CAPTEUR (varIN : SCL, SDA, CH, DA  varOUT : TT, Tf) -----
LECTURE:
Channel:
    Gosub  I2C_START
           Shiftout SCL,SDA,2,&H30      'AD,0 adresse I2C (DS2482) ecrire
           Shiftout SCL,SDA,2,&HC3      'CHSL  Pointer sur Channel Selection Register

           Shiftout SCL,SDA,2,CH      'CHANNEL choisi
    Gosub  I2C_STOP

```

# test-carte8.bas

```

Gosub OnwReset
Gosub OWWread

CONVER:
  DA=&HCC : Gosub OWWrite
  DA=&H44 : Gosub OWWrite
  Gosub BUSY
  Gosub OWWread
  Sig=KH AND &H80
  Tf=KL AND &H0F
  TL=KL AND &HF0 : TL=TL/16
  TH=KH AND &H07 : TH = TH*16
  TT=TL+TH
  if Sig = 1 then TT=&H80-TT

return

```

```

'-----'ROUTINE ECRITURE DS2482-800 1-WIRE (varIN : SCL, SDA, DA) -----
OWWrite:

```

```

  Gosub I2C_START
  Shiftout SCL,SDA,2,&H30      'AD,0 adresse I2C (DS2482)ecrire
  Shiftout SCL,SDA,2,&HA5      'command 1WWB 1wire write byte
  Shiftout SCL,SDA,2,DA       'Envoie variable
  Gosub I2C_STOP

```

```

return

```

```

'----- ROUTINE LECTURE DS2482-800 1-WIRE (varIN : SCL, SDA, DA, varOUT : KL, KH) -----
OWWread:

```

```

  DA=&HCC : Gosub OWWrite      'Commande skip ROM > DS18B20
  DA=&HBE : Gosub OWWrite      'Commande Lecture > DS18B20
  Gosub I2C_START
  Shiftout SCL,SDA,2,&H30      'AD,0 Adresse Ecriture DS2482
  Shiftout SCL,SDA,2,&H96      '1WRB Lecture byte > 1-wire
  Gosub I2C_START
  Shiftout SCL,SDA,2,&H30      'AD,0 Adresse Ecriture DS2482
  Shiftout SCL,SDA,2,&HE1      'SRP Set Read Pointer (E1)
  Shiftout SCL,SDA,2,&HE1      'E1h Pointer sur Read Data Register
  Gosub I2C_START
  Shiftout SCL,SDA,2,&H31      'AD,1 adresse lecture (DS2482)
  KL=Shiftin (SCL,SDA,1)      'Byte > KL
  Gosub I2C_START
  Shiftout SCL,SDA,2,&H30      'AD,0 Adresse Ecriture DS2482
  Shiftout SCL,SDA,2,&H96      '1WRB Lecture byte > 1-wire

```

```

                                test-carte8.bas
                                'AD,0
                                'SRP    Set Read Pointer
                                'E1h    Pointer sur Read Data Register
                                'AD,1 adresse lecture(DS2482)
                                'Byte > KH
                                '*** INDISPENSABLE ***
                                '----- RESET DS2482-800 1-WIRE -----
                                '----- TEST LIGNE 1-WIRE (varIN : SCL, SDA) -----
                                '----- START I2C (varIN : SCL, SDA) -----

Gosub I2C_START
  Shiftout SCL,SDA,2,&H30
  Shiftout SCL,SDA,2,&HE1
  Shiftout SCL,SDA,2,&HE1
Gosub I2C_START
  Shiftout SCL,SDA,2,&H31
  KH=Shiftin (SCL,SDA,1)
Gosub I2C_STOP
Gosub BUSY

return

OneWReset:
  Gosub I2C_START
    Shiftout SCL,SDA,2,&H30
    Shiftout SCL,SDA,2,&HB4
  Gosub I2C_STOP

return

BUSY:
  Gosub I2C_START
    Shiftout SCL,SDA,2,&H30
    Shiftout SCL,SDA,2,&HB4
    delay 10
  Gosub I2C_START
    Shiftout SCL,SDA,2,&H31
  RE:
    BY = Shiftin (SCL,SDA,1)
    BY = BY AND &b00000001
    IF BY = 1 THEN
      CLS
      Locate 0,0
      Print "WAIT BY=",HEX(BY,2,1)
      GOTO RE
    END IF
  gosub I2C_STOP

return

I2C_START:
  Out SCL,1

```

# test-carte8.bas

```

                                Out SDA,1
                                Out SDA,0
return

'----- STOP I2C (varIN : SCL, SDA) ----
I2C_STOP:
                                Out SDA,0
                                Out SCL,1
                                Out SDA,1
return

'----- START CIRCULATEUR (varIN : V1, V3, tempov3, tempovar) ----
CIRC_START:
                                etatCIRC = 1                                'flag etat circulateur
                                Tchi = Tch                                'memorisation temp initiale
                                Tmax = Tch                                'memorisation pour la temp max
                                out SV1,0                                'arret commande vitesse 1
                                delay 100                                'attente pour eviter double commande
                                out SV3,1                                'commande vitesse 3
                                out LEDV,0                                'arret commande témoin circulateur au repos
                                out LEDJ,1                                'debut commande témoin vitesse 3
                                tempovar = tempov3                        'temporisation
                                GOSUB TEMPO1S
                                out SV3,0                                'arret vitesse 3
                                out LEDJ,0                                'arret commande témoin vitesse 3
                                delay 100                                'attente pour eviter double commande
                                out SV1,1                                'commande vitesse 1
                                out LEDR2,1                                'debut commande témoin vitesse 1
return

'----- STOP CIRCULATEUR (varIN : V1, V3) ----
CIRC_STOP:
                                etatCIRC = 0                                'flag etat circulateur
                                LOCATE 0,1
                                PRINT "                                "                                'arret affichage retour eau capteur
                                out LEDJ,0                                'arret commande témoin vitesse 3
                                out LEDR2,0                                'arret commande témoin vitesse 1
                                out LEDV,1                                'debut commande témoin circulateur au repos
                                out SV1,0                                'arret vitesse 1
                                out SV3,0                                'arret vitesse 3
return

```

# test-carte8.bas

```
'----- temporisation (varIN : tempovar)-----
```

```
TEMPO1S:
```

```
    for j = 1 to tempovar
        delay 1000
    next j
```

```
return
```

```
'----- affichage/extinction du témoin de fonctionnement du circulateur (varIN : Tch, Tchmin)-----
affichETAT:
```

```
    '---clignotement LEDR1 si temp haut cuve insuffisante
```

```
    if Tch < Tchmin then
        out LEDR1,1
        delay 30
        out LEDR1,0
        delay 120
        out LEDR1,1
        delay 30
        out LEDR1,0
```

```
    else
        out LEDR1,0
```

```
    end if
```

```
    '---clignotement LEDs si mode manu
```

```
    if autoManu = 0 then
```

```
        if etatCIRC = 1 then
```

```
            out LEDJ,1
            out LEDV,1
            delay 30
            out LEDJ,0
            out LEDV,0
            delay 120
            out LEDJ,1
            out LEDV,1
            delay 30
            out LEDJ,0
            out LEDV,0
```

```
        else
```

```
            out LEDJ,1
```

test-carte8.bas

```
        out LEDR2,1
        delay 30
        out LEDJ,0
        out LEDR2,0
        delay 120
        out LEDJ,1
        out LEDR2,1
        delay 30
        out LEDJ,0
        out LEDR2,0
    end if
else
    if etatCIRC = 1 then
        out LEDR2,1
        out LEDJ,0
        out LEDV,0
    else
        out LEDR2,0
        out LEDJ,0
        out LEDV,1
    end if
end if
return
```

'----- affichage et choix du mode de gestion

affichMODE:

```
'-----
'--- choix du mode ---
'-----
'affichage fond
CLS
locate 0,0
print "Gestion Solaire v"
locate 17,0
print dec(version,2,0)
locate 0,2
print "Mode automatique"
locate 0,3
print "Mode manuel"
```

# test-carte8.bas

```
'affichage de la situation actuelle
if autoManu = 1 then
    locate 19,2
else
    locate 19,3
end if
print "*"
```

```
'test des bp : bleu changement, rouge validation
autoManuTemp = autoManu
for n = 1 to Nmax
    bouton = keyin(bpBleu,50)
    if bouton = 0 then
        if autoManuTemp = 1 then
            autoManuTemp = 0
            locate 19,2
            print " "
            locate 19,3
            print "*"
        else
            autoManuTemp = 1
            locate 19,2
            print "*"
            locate 19,3
            print " "
        end if
    end if
    bouton = keyin(bpRouge,50)
    if bouton = 0 then
        n = Nmax
    end if
end if
next n
```

```
'-----
'--- menu des differents modes ---
'-----

'--- affichage menu mode manuel ---
if autoManuTemp = 0 then
    CLS
    locate 0,0
    print "Mode Manuel"
```

```
'prise en compte de l etat precedent
'test bp bleu (changement)
'ancien = auto
'nouveau = manuel

'ancien = manuel, nouveau = auto

'test bp rouge (validation)
'fin de la boucle
```

test-carte8.bas

```
locate 0,1
print "Circulateur :"
if etatCIRC = 1 then
    locate 18,1
    print "ON"
    locate 0,2
    print "-> Arrêter (rouge)"
    locate 0,3
    print "-> Retour (bleu)"
else
    locate 17,1
    print "OFF"
    locate 0,2
    print "-> Démarrer (rouge)"
    locate 0,3
    print "-> Retour (bleu)"
end if
```

```
'test des boutons
etatCircTemp = 2
for n = 1 to Nmax
    bouton = keyin(bpBleu,50)
    if bouton = 0 then
        n = Nmax
    end if

    bouton = keyin(bpRouge,50)
    if bouton = 0 then
        if etatCIRC = 1 then
            etatCircTemp = 0
        else
            etatCircTemp = 1
        end if
    end if
    n = Nmax
end if
next n

'affichage prise en compte bouton
locate 17,0
print "----"
delay 1000
```

'init de la valeur (différente de 0 ou 1)

'test bp bleu (cancel)

'test bp rouge (changement)

'memo etat

'memo etat



test-carte8.bas

```
'confirmation
CLS
locate 0,0
print "Mode Manuel"
locate 0,2
if etatCircTemp = 0 then
    print "Arret en cours.."
elseif etatCircTemp = 1 then
    print "Demarrage en cours.."
elseif etatCircTemp = 2 then
    print "Aucun changement"
end if
delay 2000

'commande
if etatCircTemp = 0 then
    GOSUB CIRC_STOP
elseif etatCircTemp = 1 then
    GOSUB CIRC_START
end if

else
    'mode automatique
    'confirmation
    CLS
    locate 0,0
    print "Mode Automatique"
    locate 0,2
    if autoManu = 1 then
        print "Enregistre"
    else
        print "Changement OK"
    end if
    delay 2000

end if

'enregistrement valeur
autoManu = autoManuTemp

return
```

'precedent auto : pas de changement

'precedent manu

test-carte8.bas