```
1   ; ********************************************************************************************************************
2   ; **  PROGRAMME MAKED BY FADISHOP   Card: FADICLOCK   http://www.fadishop.eu  FADITECO, S.L.U.  Lleonard Garcia   **
3   ; **  This program reads the temperature every x seconds, averaged every minute and stored in the SRAM            **
4   ; **  Activated alarm 1 (every second) and Alarm 2 (every minute) connectadas to interrupt INT2 (B.2)             **
5   ; **  SLAVE ADDRESS IS STILL DS3232 I2C $ D0.                                                                     **
6   ; **  INTERRUPT ROUTING / RST JP6 (Blue jumper) (I0=1, I1=0, I2=1, portC.7=0;  Rpullup=1)                         **
7   ; ********************************************************************************************************************
8   ; SETTINGS
9   #picaxe 28x2
10  let dirsB=%10000000           ; 1=output 0=input
11  let dirsC=%00000000           ; 1=output 0=input
12  ; C.3 I2C_SCL
13  ; C.4 I2C_SDA
14  ;adcsetup = %zy0000000000xxxx ; SETTING ANALOG(z=0-GND, y=0-Vcc   xxxx inputs adc 0000 0001 0011 0111 1111)
15  ;setfreq em16                 ; Oscillator / external oscillator to 16Mhz.
16  device_DS3232SN:
17  symbol adress_slave_A =$D0    ; Address 0xD0 DS3232 I2C only.
18  symbol @now=$00               ; Internal adress register of seconds, minutes and hours.
19  symbol @today=$03             ; Internal adress register of day, month and year-century.
20  symbol @alm1=$07              ; Internal adress register alarm_1 (sec, min, hour and day / day).
21  symbol @alm2=$0B              ; Internal adress register alarm2 (min, hour and day / day).
22  symbol @control=$0E           ; (7)/EOSC (6)BBSQW (5)CONV (4-3)RS2-1 (2)INTCN (1)A2IE (0)A1IE
23  symbol @status=$0F            ; (7)OSF (6)BB32KHZ (5-4)CRATE (3)EN32KHZ (2)BSY (1)A2F (0)A1F
24  symbol @offset=$10            ; temperature offset. (adds / subtracts capacitance)
25  symbol @temp=$11              ; Internal word temperature address($11-MSB_temperature and $12-LSB_temperature) .
26  symbol @test=$13              ; Reserved for test.
27  symbol @sram=$14              ; Internal address of the start of the block of 236 bytes SRAM.
28  symbol control=b16            ; Register/shadow control DS3232.
29  symbol status=b18             ; Register/shadow control/status DS3232.
30  ; Program Definitions and Initialization
31  symbol seconds=b0
32  symbol minutes=b1
33  symbol hour=b2
34  symbol day=b3
35  symbol date=b4
36  symbol mounth=b5
37  symbol year=b6
38  symbol T_msb=b7
39  symbol T_lsb=b8
40
41  symbol acumMsb=w19
42  symbol acumLsb=w20
43  symbol medMinMsb=w21
44  symbol medMinLsb=w22
45  symbol N_sample=b48
46  symbol @SramMin=$20                     ; address of the SRAM that stores the average
47
48  ;INICIALITZACIONES:
49  let control =%00011100                  ; match default initialization.
50  let status =%11001000                   ; match default initialization.
```

```
51   call set_date_time                              ; change the date-time DS3232 RTC.
52   call set_alarm1                                 ; sets alarma_1 in the DS3232 RTC .
53   call set_alarm2                                 ; sets alarma_2 in the DS3232 RTC.
54   interrupt_settings:
55   hint2flag = 0                                   ; clear the flag INT_2.
56   symbol hint_setting = %00000100                 ; active interrupts INT2.
57   hintsetup hint_setting                          ; transfers such activation.
58   setintflags %00000100,%00000100                 ; INT2 interrupts are enabled per falling edge.
59
60   main:
61   call  temperature_request
62   i2cslave adress_slave_A, i2cslow, i2cbyte       ; I2C address of DS3232 (0xD0-0xD1).
63   readi2c @now,(seconds,minutes,hour,day)         ; reading time
64   ;readi2c @today,(date,mounth,year)              ; reading date.
65   readi2c @temp,(T_msb,T_lsb)                     ; temperature reading.
66   ;readi2c @alm1,(b10,b11,b12,b13)                ; alarm_1 reading.
67   ;readi2c @control,(b17,b19)                     ; reading register.
68   ;readi2c INICI,(b0,b1)
69   debug
70   PAUSE 2000
71   goto main
72
73
74   ; INTERRUPT SEQUENCE
75   interrupt:
76   readi2c @status,(status)                        ; It reads the control and status registers.
77   let b29=status and %00000001
78   if b29!=0 then call alarm1                      ; If A1F = 1 runs Alarm1.
79   let b29=status and %00000010
80   if b29!=0 then call alarm2                      ; Yes A2F = 1 runs Alarm2.
81   hint2flag = 0                                   ; Re-enable interrupts INT2.
82   hintsetup hint_setting                          ; Re-enable interrupts INT2.
83   setintflags %00000100,%00000100                 ; Interrupts are enabled and INT2 INT1 by falling edge.
84   return                                          ; After return is when you actually enable the interrupt.
85
86   alarm1:
87   let status=status and %11111110                 ; Clears the flag on DS3232 alarma_1.
88   writei2c @status,(status)                       ; Enable and delete flag A1F.
89   call  add_sample
90   return
91
92
93   alarm2:
94   let status=status and %11111101                 ; Clears the flag on DS3232 alarma_2.
95   writei2c @status,(status)                       ; Enable and delete flag A2F.
96   call seconds_average
97   let acumMsb=0
98   let acumLsb=0
99   let N_sample=0
100
```

```
101  let b50=@SramMin
102  let b51=bcdtobin minutes
103  let b51=b51*2
104  let b50=b50 + b51
105  writei2c b50,(medMinMsb,medMinLsb)
106  readi2c @SramMin,(b52,b53,b54,b55)
107  return
108
109
110  set_date_time:
111  symbol xseg=b30
112  symbol xminu=b31
113  symbol xhour=b32
114  symbol xsemdate=b33
115  symbol xdate=b34
116  symbol xmounth=b35
117  symbol xyear=b36
118  let xseg=$25                                    ; @_internal DS3232: $00
119  let xminu=$00                                   ; @_internal DS3232: $01
120  let xhour=$13                                   ; @_internal DS3232: $02
121  let xsemdate=$00                                ; @_internal DS3232: $03
122  let xdate=$03                                   ; @_internal DS3232: $04
123  let xmounth=$03                                 ; @_internal DS3232: $05
124  let xyear=$12                                   ; @_internal DS3232: $06
125  low xhour,6                                     ; LOW: 24H          HIGH: AM/PM
126  i2cslave adress_slave_A, i2cslow, i2cbyte       ; Dirección I2C del DS3232 (0XD0-0XD1).
127  writei2c @now,(xseg,xminu,xhour,xsemdate,xdate,xmounth,xyear); I2C address of DS3232 (0xD0-0xD1)
128  return
129
130
131  set_alarm1:
132  symbol a1_seg=b30
133  symbol a1_min=b31
134  symbol a1_hor=b32
135  symbol a1_day=b33
136  symbol a1_mode=b34
137  let a1_seg=$00
138  let a1_min=$01
139  let a1_hor=$13                                  ; Time Format->bit(6)=0:24h, bit(6)=1:AM/PM 12h.
140  let a1_day=$03                                  ; Date Format-> bit(6)=0:date del mounth, bit(6)=1:date de la semana.
141  let a1_mode=4
142  low a1_hor,6                                    ; LOW: 24H              HIGH: AM/PM
143  low a1_day,6                                    ; LOW: date             HIGH: day of week.
                                    ;
144  select case a1_mode
145  case 0
146  clearbit a1_seg,7  clearbit a1_min,7  clearbit a1_hor,7   clearbit a1_day,7   ;0: exactly (date, hours, minutes,
     second).
147  case 1
148  clearbit a1_seg,7  clearbit a1_min,7  clearbit a1_hor,7   setbit a1_day,7     ;1: exactly (hours, minutes, seconds).
```

```
149  case 2
150  clearbit a1_seg,7  clearbit a1_min,7  setbit a1_hor,7  setbit a1_day,7        ;2: exactly (minutes, seconds).
151  case 3
152  clearbit a1_seg,7  setbit a1_min,7 setbit a1_hor,7  setbit a1_day,7           ;3: exactly (seconds).
153  case 4
154  setbit a1_seg,7 setbit a1_min,7 setbit a1_hor,7  setbit a1_day,7             ;4: periodic generator 1 second.
155  end select
156
157  i2cslave adress_slave_A, i2cslow, i2cbyte       ; I2C address of DS3232 (0xD0-0xD1).
158  writei2c @alm1,(a1_seg,a1_min,a1_hor,a1_day)    ; Send configuration of alarm_1.
159  let control=control or %00000001                ; Enabled interrupt of alarm_1 A1E.
160  let control=control or %00000100                ; Enable interrupt alarm_1 in DS3232 INTCN.
161  let status=status and %11111110                 ; Clears flag A1F on DS3232 alarm_1.
162  writei2c @control,(control,status)              ; Send status.
163  return
164
165  set_alarm2:
166  symbol a2_min=b31
167  symbol a2_hor=b32
168  symbol a2_day=b33
169  symbol a2_modo=b34
170  let a2_min=$01
171  let a2_hor=$13                                  ; Time format->bit(6)=0:24h,        bit(6)=1:AM/PM 12h.
172  let a2_day=$03                                  ; Date format-> bit(6)=0:date,      bit(6)=1:day of week.
173  let a2_modo=4
174  low a2_hor,6                                    ; LOW: 24H          HIGH: AM/PM
175  low a2_day,6                                    ; LOW: date         HIGH: day of week.
                                    ;
176  select case a2_modo
177  case 0
178  clearbit a2_min,7  clearbit a2_hor,7   clearbit a2_day,7    ;0: exactly (date, hours, minutes).
179  case 1
180  clearbit a2_min,7  clearbit a2_hor,7   high a2_day,7        ;1: exactly (hours, minutes).
181  case 2
182  clearbit a2_min,7  setbit a2_hor,7  setbit a2_day,7         ;2: exactly (minutes).
183  case 3
184  setbit a2_min,7 setbit a2_hor,7  setbit a2_day,7            ;3: periodic generator 1 second.
185  case 4
186  setbit a2_min,7 setbit a2_hor,7  setbit a2_day,7            ;4: periodic generator 1 second.
187  end select
188
189
190  i2cslave adress_slave_A, i2cslow, i2cbyte       ; I2C address of DS3232 (0xD0-0xD1).
191  writei2c @alm2,(a2_min,a2_hor,a2_day)           ; Send configuration of alarm_2.
192  let control=control or %00000010                ; Enabled interrupt of alarm_2 A2E.
193  let control=control or %00000100                ; Enable interrupt alarm_2 in DS3232 INTCN.
194  let status=status and %11111101                 ; Clears flag A2F on DS3232 alarm_2.
195  writei2c @control,(control,status)              ; Send status.
196  return
197
```

```
198
199  add_sample:
200  acumMsb=T_msb+acumMsb
201  acumLsb=T_lsb+acumLsb
202  inc N_sample
203  return
204
205  seconds_average:
206  medMinMsb=acumMsb/N_sample
207  medMinLsb=acumLsb/N_sample
208  return
209
210  temperature_request:
211  i2cslave adress_slave_A, i2cslow, i2cbyte       ; I2C address of DS3232 (0xD0-0xD1).
212  readi2c @status,(b10)                           ; Read status byte of DS3232
213  let b10=b10 and %00000100                       ; Is performing a conversion (BSY=1)?
214  if b10<>0 then return                           ; (YES=1) Do nothing, come back.
215            else let control=control or %00100000 ; (NO=0) Force a new temperature conversion.
216                    i2cslave adress_slave_A, i2cslow, i2cbyte       ;I2C address of DS3232 (0xD0-0xD1).
217                    writei2c @control,(control)
218  endif
219  return
220
```