

```

1 ; *****
2 ; ** REALIZADO POR FADISHOP Tarjeta: FADICLOCK      http://www.fadishop.eu FADITECO, S.L.U. Lleonard Garcia  **
3 ; ** Este programa lee la hora actual y saca por un cuadruple display lahora-minuto o minutos-segundos  **
4 ; ** LA DIRECCIÓN DEL SLAVE I2C DS3232 ES FIJA $D0.  **
5 ; ** ENCAMINAMIENTO INTERRUPCIÓN ALARMAS JP6(JUMPER AZUL) (I0=1, I1=0, I2=0, portC.7=0; Rpullup=0)  **
6 ; **
7 ; *****
8 ; CONFIGURACIONES
9 #picaxe 28x2
10 let dirsB=%11111111 ; 1=salida 0=entrada
11 let dirsC=%11111111 ; 1=salida 0=entrada
12 ; C.3 I2C_SCL
13 ; C.4 I2C_SDA
14 ;adcsetup = %zy000000000xxxxx ; SETTING ANALOG(z=0-GND, y=0-Vcc  xxxxx inputs adc 0000 0001 0011 0111 1111)
15 setfreq eml6 ; Resonador/oscilador externo a 16Mhz.
16 device_DS3232SN:
17 symbol adress_slave_A =$D0 ; Dirección única I2C del DS3232 0XD0
18 symbol @ahora=$00 ; dirección interna de segundos, minutos y hora.
19 symbol @hoy=$03 ; dirección interna de dia_semanal, dia, siglo-mes y año.
20 symbol @alm1=$07 ; dirección interna de alarma_1(seg, min, hora y dia_sem/dia).
21 symbol @alm2=$0B ; dirección interna de alarma_2(min, hora y dia_sem/dia).
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 ; offset temperatura. (añade/sustrae capacitancia)
25 symbol @temp=$11 ; dirección word interna de MSB_temperatura y LSB_temperatura.
26 symbol @test=$13 ; reservada par test.
27 symbol @sram=$14 ; dirección interna del inicio del bloque de 235 bytes SRAM.
28 symbol control=b16 ; variable control DS3232.
29 symbol status=b18 ; variable control/status DS3232.
30 ; DEFINICIONES Y INICIALIZACIONES de programa
31 symbol segundos=b0
32 symbol minutos=b1
33 symbol hora=b2
34 symbol dia_sem=b3
35 symbol dia=b4
36 symbol mes=b5
37 symbol anno=b6
38 symbol T_msb=b7
39 symbol T_lsb=b8
40
41 symbol display_1=b21
42 symbol display_2=b22
43 symbol display_3=b23
44 symbol display_4=b24
45 symbol wreg=b25
46 symbol turno=b26
47 symbol tarea=b27
48 symbol isplay_1=b31
49 symbol isplay_2=b32
50 symbol isplay_3=b33

```

```
51 symbol isplay_4=b34
52
53 call tarea_i2c
54
55 main:
56 if tarea>254 then
57     call tarea_i2c
58     let tarea=0
59     else
60         call refrescar
61         inc tarea
62 end if
63 ;debug
64 goto main
65
66 tarea_i2c:
67 i2cslave adress_slave_A, i2cslow, i2cbyte           ; Dirección I2C del DS3232 (0XD0-0XD1)
68 readi2c $00,(segundos,minutos)                     ; *1.-Minutos:segundos
69 bcdtoascii segundos,display_3,display_4             ; *1.-Minutos:segundos
70 bcdtoascii minutos,display_1,display_2              ; *1.-Minutos:segundos
71 ;readi2c $01,(minutos,hora)                         ; *2.-Hora:minutos
72 ;bcdtoascii minutos,display_3,display_4             ; *2.-Hora:minutos
73 ;bcdtoascii hora,display_1,display_2                ; *2.-Hora:minutos
74 let wreg=display_1-$30
75 call dec7seg
76 let display_1=wreg
77 let wreg=display_2-$30
78 call dec7seg
79 let display_2=wreg
80 let wreg=display_3-$30
81 call dec7seg
82 let display_3=wreg
83 let wreg=display_4-$30
84 call dec7seg
85 let display_4=wreg
86 return
87
88
89 refrescar:
90 high C.0, C.1, C.2, C.6
91 select case turno
92 case 0 to 1
93 let turno=turno+1
94 let pinsB=display_1
95 low C.0
96 case 2
97 let turno=turno+1
98 let pinsB=display_2
99 low C.1
100 case 3
```

```
101 let turno=turno+1
102 let pinsB=display_3
103 low C.2
104 case 4
105 let turno=1
106 let pinsB=display_4
107 low C.6
108 endselect
109 return
110
111
112
113 dec7seg:
114 select case wreg
115 case 0
116 ;let wreg=%01111110
117 let wreg=%10000001 ;lógica negativa anodo común transistor PNP
118 case 1
119 ;let wreg=%00000110
120 let wreg=%11111001 ;lógica negativa anodo común transistor PNP
121 case 2
122 ;let wreg=%11110010
123 let wreg=%00001101 ;lógica negativa anodo común transistor PNP
124 case 3
125 ;let wreg=%11010110
126 let wreg=%00101001 ;lógica negativa anodo común transistor PNP
127 case 4
128 ;let wreg=%10001110
129 let wreg=%01110001 ;lógica negativa anodo común transistor PNP
130 case 5
131 ;let wreg=%11011100
132 let wreg=%00100011 ;lógica negativa anodo común transistor PNP
133 case 6
134 ;let wreg=%11111100
135 let wreg=%00000011 ;lógica negativa anodo común transistor PNP
136 case 7
137 ;let wreg=%00010110
138 let wreg=%11101001 ;lógica negativa anodo común transistor PNP
139 case 8
140 ;let wreg=%11111110
141 let wreg=%00000001 ;lógica negativa anodo común transistor PNP
142 case 9
143 ;let wreg=%11011110
144 let wreg=%00100001 ;lógica negativa anodo común transistor PNP
145 endselect
146 return
```