

```

1 ; *****
2 ; ** PROGRAMA REALIZADO POR FADISHOP Tarjeta: FADIBUS http://www.fadishop.es FADITECO,S.L.U. **
3 ; ** El SLAVE PCA9555N_A[$42] SUPLICA A LA CPU POR INT0, El SLAVE PCA9555N_B[$44] SUPLICA A LA CPU POR INT1 **
4 ; ** SLAVE [$42]_A: LEE SU PUERTO_0 Y SACA EL DATO LEIDO POR SU PUERTO_1 DE FORMA INMEDIATA **
5 ; ** SLAVE [$44]_B: LEE SU PUERTO_0 Y SACA EL DATO LEIDO POR SU PUERTO_1 DE FORMA INMEDIATA **
6 ; ** LA DIRECCIÓN DEL SLAVES I2C ES $42 --> SW2[$42](A=0, B=0, C=1) SW1(I0=1, I1=0, I2=0, portC=0, Rpullup=1) **
7 ; ** LA DIRECCIÓN DEL SLAVES I2C ES $44 --> SW2[$44](A=0, B=1, C=0) SW1(I0=0, I1=1, I2=0, portC=0, Rpullup=1) **
8 ; ** JP9(no afecta JUMPER AZUL) **
9 ; ** PRECAUCIÓN: CON UNA RESISTENCIA DE PULLUP POR INTERRUPCIÓN ES SUFICIENTE.(Rs o Rpullup) **
10 ; *****
11 ; CONFIGURACIONES
12 #picaxe 28x2
13 let dirsB=%00000000 ; 1=salida 0=entrada
14 let dirsC=%00000000 ; 1=salida 0=entrada
15 ; C.3 I2C_SCL
16 ; C.4 I2C_SDA
17 ;adcsetup = %00000000 ' CONFIGURACIÓN ENTRADAS ANALÓGICAS
18 ;setfreq em16 ; Resonador/oscilador externo a 16Mhz.
19
20 device_PCA9555N: ;DIRECCIONES INTERNAS DEL EXPANSOR DE BUS PCA9555N
21 symbol inport_0 = 0 ; lectura de entradas del puerto_0
22 symbol inport_1 = 1 ; lectura de entradas del puerto_1
23 symbol outport_0 = 2 ; escritura en salidas del puerto_0.
24 symbol outport_1 = 3 ; escritura en salidas del puerto_1.
25 symbol polaritat_port_0 = 4 ; Pol·laridad entradas puerto_0 --> 0=no_invertido, 1=invertido.
26 symbol polaritat_port_1 = 5 ; Pol·laridad entradas puerto_1 --> 0=no_invertido, 1=invertido.
27 symbol config_port_0 = 6 ; configura puerto_0 como entrada o salida 1=entrada 0=salida.
28 symbol config_port_1 = 7 ; configura puerto_1 como entrada o salida 1=entrada 0=salida.
29 config_PCA9555N_A: ;CONFIGURACIÓN DEL DEVICE_A (A,B,C,D,E,F,G,H. M̀ximo 8)
30 symbol adress_slave_A = $42 ; Direcci3n I2C del PCA9555N % 0100ABC(rw) ;SLAVE PCA9555N (0X4E-0X4F)
31 symbol polport0_A = %00000000 ; polaridad de las entradas del puerto_0: 0-sin inversi3n 1-invertidas
32 symbol polport1_A = %00000000 ; polaridad de las entradas del puerto_1: 0-sin inversi3n 1-invertidas
33 symbol confport0_A = %11111111 ;IN ; configuraci3n puerto_0 1=entrada 0=salida.
34 symbol confport1_A = %00000000 ;OUT ; configuraci3n puerto_1 1=entrada 0=salida.
35 config_PCA9555N_B: ;CONFIGURACIÓN DEL DEVICE_B (A,B,C,D,E,F,G,H. M̀ximo 8)
36 symbol adress_slave_B = $44 ; Direcci3n I2C del PCA9555N % 0100ABC(rw) ;SLAVE PCA9555N (0X4E-0X4F)
37 symbol polport0_B = %00000000 ; polaridad de las entradas del puerto_0: 0-sin inversi3n 1-invertidas
38 symbol polport1_B = %00000000 ; polaridad de las entradas del puerto_1: 0-sin inversi3n 1-invertidas
39 symbol confport0_B = %11111111 ;IN ; configuraci3n puerto_0 1=entrada 0=salida.
40 symbol confport1_B = %00000000 ;OUT ; configuraci3n puerto_1 1=entrada 0=salida.
41 transfer_config_PCA9555N_A: ;TRANSFERIR TODA LA CONFIGURACIÓN AL DEVICE_A
42 i2cslave adress_slave_A, i2cslow, i2cbyte ; envia/llamada a direcci3n del perif3rico/slave.
43 writei2c polaritat_port_0,(polport0_A,polport1_A); envia polaridad de las entradas del puerto_0 y puerto_1
44 writei2c config_port_0,(confport0_A,confport1_A); envia configuraci3n del puerto_0 y puerto_1.
45 transfer_config_PCA9555N_B: ;TRANSFERIR TODA LA CONFIGURACIÓN AL DEVICE_B
46 i2cslave adress_slave_B, i2cslow, i2cbyte ; envia/llamada a direcci3n del perif3rico/slave.
47 writei2c polaritat_port_0,(polport0_B,polport1_B); envia polaridad de las entradas del puerto_0 y puerto_1
48 writei2c config_port_0,(confport0_B,confport1_B); envia configuraci3n del puerto_0 y puerto_1.
49 configuracio_interrupcions:
50 symbol hint_setting = %00000011 ; se activan las interrupciones por INT0 y INT1.

```

```

51 call rearmar_interrup           ; CONTINUA. Secuencia común compartida ubicada en interrupt.
52 ; INICIO
53 ; DEFINICIONES Y INICIALIZACIONES
54 main:                           ; PROGRAMA CÍCLICO.
55 debug                           ; Saca por una ventana emergente todos los registros.Añade retardo..
56 goto main                       ; volver a empezar.
57
58 ; SECUENCIA DE INTERRUPCIÓN
59 interrupt:                       ; sólo se atiende al puerto de entrada que ha cambiado.
60 atenINT0:
61 if hint0flag=0 then atenINT1     ; Interrupción INT0=1? (NO) SALTAR
62 i2cslave adress_slave_A, i2cslow, i2cbyte ; (SI) envia/llamada a dirección del periférico/slave.
63 readi2c inport_0,(b0)           ; se lee el puerto de entrada          inport_0 --> b0
64 writei2c outport_1,(b0)        ; se escribe lo leído en el puerto_1    b0 --> outport_1
65
66 atenINT1:
67 if hint1flag=0 then rearmar_interrup ; Interrupción INT1=1? (NO) SALTAR
68 i2cslave adress_slave_B, i2cslow, i2cbyte ; (SI) envia/llamada a dirección del periférico/slave.
69 readi2c inport_0,(b0)           ; se lee el puerto de entrada          inport_0 --> b0
70 writei2c outport_1,(b0)        ; se escribe lo leído en el puerto_1    b0 --> outport_1
71
72 rearmar_interrup:
73 hint0flag = 0                   ; borrar el flag INT0.
74 hint1flag = 0                   ; borrar el flag INT1.
75 hintsetup hint_setting          ; se vuelven a activar las interrupciones por INT0 o INT1
76 setintflags or %00000011,%00000011 ; se habilitan interrupciones por INT0 o INT1 en flanco descendente.
77 return                          ; después de return es cuando realmente habilitan las interrupciones
78
79
80

```