

```

1
2 ; *****
3 ; ** PROGRAM MADE BY FADISHOP Card: FADIBUS http://www.fadishop.eu FADITECO, S.L.U. Lleonard Garcia **
4 ; ** The SLAVE PCA9555N_A [$42] APPEAL BY INTO TO CPU. The SLAVE PCA9555N_B [$44] APPEAL BY INTO TO CPU. **
5 ; ** SLAVE [$42]_A: READ YOUR PORT 0, IMMEDIATELY IT TAKES FOR THE PORT_1. **
6 ; ** SLAVE [$44]_B: READ YOUR PORT 0, IMMEDIATELY IT TAKES FOR THE PORT_1. **
7 ; ** I2C SLAVE ADDRESS IS $ 42--> SW2[$42](A=0, B=0, C=1) SW1(I0=1, I1=0, I2=0, portC=0, Rpullup=1) **
8 ; ** I2C SLAVE ADDRESS IS $ 44--> SW2[$42](A=0, B=1, C=0) SW1(I0=0, I1=1, I2=0, portC=0, Rpullup=0) **
9 ; ** JP9 (does not affect JUMPER BLUE) **
10 ; ** CAUTION: WITH A ONE PULLUP RESISTOR FOR LINE INTERRUPT IS ENOUGH. (Rs or Rpullup) **
11 ; *****
12 #picaxe 28x2
13 ; SETTINGS
14 let dirsB=%00000000 ; 1=output 0=input
15 let dirsC=%00000000 ; 1=output 0=input
16 ; C.3 I2C_SCL
17 ; C.4 I2C_SDA
18 ;adcsetup = %00000000 ; SETTING ANALOG
19 ;setfreq em16 ; Oscillator / external oscillator to 16Mhz.
20 ; DEFINITIONS AND INITIALIZATIONS DEVICES.
21 device_PCA9555N: ;INTERNAL ADDRESS BUS EXPANDER PCA9555N
22 symbol inport_0 = 0 ; Read register inputs port_0.
23 symbol inport_1 = 1 ; Read register inputs port_1.
24 symbol outport_0 = 2 ; Write register outputs port_0.
25 symbol outport_1 = 3 ; Write register outputs port_0.
26 symbol polaritat_port_0 = 4 ; Polarity register port_0 inputs -> 0 = no_inverted, 1 = inverted.
27 symbol polaritat_port_1 = 5 ; Polarity register port_1 inputs -> 0 = no_inverted, 1 = inverted.
28 symbol config_port_0 = 6 ; Configuration register port_0 as input or output. 1=input 0= utput.
29 symbol config_port_1 = 7 ; Configuration register port_1 as input or output. 1=input 0= utput.
30 ; SETTINGS
31 config_PCA9555N_A: ;DEVICE_A SETUP (A, B, C, D, E, F, G, H. Maximum 8)
32 symbol adress_slave_A =$42 ; I2C Address PCA9555N %0100ABC(rw). SLAVE PCA9555N (0X4E-0X4F).
33 symbol polport0_A = %00000000 ; polarity port_0 inputs: 0-no inverted 1-logical inverted.
34 symbol polport1_A = %00000000 ; polarity port_1 inputs: 0-no inverted 1-logical inverted.
35 symbol confport0_A = %11111111 ; IN ; configuration port_0 1=input 0=output.
36 symbol confport1_A = %00000000 ; OUT ; configuration port_0 1=input 0=output.
37 config_PCA9555N_B: ;DEVICE_B SETUP (A, B, C, D, E, F, G, H. Maximum 8)
38 symbol adress_slave_B =$44 ; I2C Address PCA9555N %0100ABC(rw). SLAVE PCA9555N (0X4E-0X4F).
39 symbol polport0_B = %00000000 ; polarity port_0 inputs: 0-no inverted 1-logical inverted.
40 symbol polport1_B = %00000000 ; polarity port_1 inputs: 0-no inverted 1-logical inverted.
41 symbol confport0_B = %11111111 ; IN ; configuration port_0 1=input 0=output.
42 symbol confport1_B = %00000000 ; OUT ; configuration port_0 1=input 0=output.
43 transfer_config_PCA9555N_A: ;TRANSFER ALL SETTINGS TO DEVICE_A.
44 i2cslave adress_slave_A, i2cslow, i2cbyte ; send/call to address the peripheral/slave.
45 writei2c polaritat_port_0,(polport0_A,polport1_A); polarity send entries and port_1 port_0
46 writei2c config_port_0,(confport0_A,confport1_A); send configuration and port_1 port_0.
47 transfer_config_PCA9555N_B: ;TRANSFER ALL SETTINGS TO DEVICE_B.
48 i2cslave adress_slave_B, i2cslow, i2cbyte ; send/call to address the peripheral/slave.
49 writei2c polaritat_port_0,(polport0_B,polport1_B); polarity send entries and port_1 port_0
50 writei2c config_port_0,(confport0_B,confport1_B); send configuration and port_1 port_0.
51 configuracio_interrupcions:

```

```

52 symbol hint_setting = %00000011           ; Are triggered interrupts INT0 and INT1.
53 call restore_interrup                     ; CONTINUES. Shared common sequence located in interrupt.
54 ;CYCLIC PROGRAM
55 main:                                     ;CYCLIC PROGRAM
56 debug                                     ; Take a pop all registros.Add delay.
57 goto main                                 ; Start over.
58 ;INTERRUPT SEQUENCE
59 interrupt:                                ; only serves the port of entry to be changed.
60 atenINT0:
61 if hint0flag=0 then atenINT1              ; Interrupt INT0=1? (NO) JUMP
62 i2cslave adress_slave_B, i2cslow, i2cbyte ; (YES) send/call to address the peripheral/slave.
63 readi2c inport_0,(b0)                    ; reads the input port.           inport_0 --> b0
64 writei2c outport_1,(b0)                  ; read what is written in the port_1 b0 --> outport_1
65 atenINT1:
66 if hint1flag=0 then restore_interrup      ; Interrupt INT1=1? (NO) JUMP
67 i2cslave adress_slave_B, i2cslow, i2cbyte ; (YES) send/call to address the peripheral/slave.
68 readi2c inport_0,(b0)                    ; reads the input port.           inport_0 --> b0
69 writei2c outport_1,(b0)                  ; read what is written in the port_1 b0 --> outport_1
70 restore_interrup:
71 hint0flag = 0                             ; clear the INT0 flag.
72 hint1flag = 0                             ; clear the INT1 flag.
73 hintsetup hint_setting                    ; re-enable interrupts INT0 or INT1
74 setintflags or %00000011,%00000011      ; interrupts are enabled by INT0 or INT1 on falling edge.
75 return

```