;CONVERSOR DE TECLADO PS2 A AMSTRAD CPC

;LA FILA DE ARRIBA DE LA PLACA DEL AMSTRAD

; TIENE 10 PINES QUE GENERAN LA SEÑAL A ESCANEAR

; Y CUENTA DESDE LA DERECHA A LA IZQUIERDA

;LA FILA DE ABAJO SON 8 PINES (LOS DE LOS EXTREMOS NO SIRVEN)

; Y CUENTA DE IZQUIERDA A DERECHA,

; QUE DEVUELVEN LA TECLA PULSADA

;DE MODO QUE SI EL PIN 1 DE LA FILA DE ARRIBA ESTÁ ACTIVO (POSITIVO)

;LA FILA DE ABAJO DEVUELVE DE IZQUIERDA A DERECHA LAS TECLAS 0 A 7

;LAS TECLAS PULSADAS DEVUELVEN UN 0 (NEGATIVO)

;SI EL PIN 2 DE LA FILA DE ARRIBA ESTÁ ACTIVO

;LA FILA DE ABAJO DEVUELVE LAS TECLAS 8 A 15 y ASI...

;PUERTOS DE PIC

;PORTB 0-7 PIN 21 AL 28 SALIDA HACIA EL AMSTRAD CON LA TECLA PULSADA

;PORTC 0-7 PIN 9 AL 18 ENTRADA DE ESCANER DE TECLADO DEL AMSTRAD (IOC)

;PORTA 6-7

;PORTA 0 PIN 2 PS2 DATA (USB D-)

;PORTA 1 PIN 3 PS2 CLK (USB D+)

;

#include "p16f18857.inc"

; CONFIG1

; \_\_config 0xD78C

\_\_CONFIG \_CONFIG1, \_FEXTOSC\_OFF & \_RSTOSC\_HFINT32 & \_CLKOUTEN\_OFF & \_CSWEN\_OFF & \_FCMEN\_OFF

; CONFIG2

; \_\_config 0xFFDC

\_\_CONFIG \_CONFIG2, \_MCLRE\_OFF & \_PWRTE\_ON & \_LPBOREN\_ON & \_BOREN\_ON & \_BORV\_LO & \_ZCD\_OFF & \_PPS1WAY\_ON & \_STVREN\_ON

; CONFIG3

; \_\_config 0xFF9F

\_\_CONFIG \_CONFIG3, \_WDTCPS\_WDTCPS\_31 & \_WDTE\_OFF & \_WDTCWS\_WDTCWS\_7 & \_WDTCCS\_SC

; CONFIG4

; \_\_config 0xCFFF

\_\_CONFIG \_CONFIG4, \_WRT\_OFF & \_SCANE\_not\_available & \_LVP\_OFF

; CONFIG5

; \_\_config 0xFFFF

\_\_CONFIG \_CONFIG5, \_CP\_OFF & \_CPD\_OFF

cblock .112

stm ;state machine for PS/2

stat ;status bits

char ;char buffer

parity ;parity byte (only one bit used)

keymatrix:.10 ;10 bytes matrix of CPC keys

next

endc

#define prev\_clk stat,0

#define data\_copy stat,1

#define release stat,2

#define extended stat,3

#define parity\_bit parity,1 ;Same bit as data\_copy

#define ps2\_clk PORTA,1

#define ps2\_data PORTA,0

#define StartBit .0

#define ParityBit .9

#define StopBit .10

#define ZERO STATUS,Z

#define CARRY STATUS,C

#define ExtKey 0xE0 ;Extended keys require 2 bytes keycode

#define Break 0xF0 ;Release of a key

org 0x0000

GOTO start ; go to beginning of program

org 0x0004

isr:

banksel LATB ;To force a faster reponse

movf next,W ;precalculate the current pulse

movwf LATB ;Have 4-6us to give the pressed keys back

banksel IOCAF

btfsc IOCAF,7

movf keymatrix+1,W

btfsc IOCAF,6

movf keymatrix+2,W

btfsc IOCCF,0

movf keymatrix+3,W

btfsc IOCCF,1

movf keymatrix+4,W

btfsc IOCCF,2

movf keymatrix+5,W

btfsc IOCCF,3

movf keymatrix+6,W

btfsc IOCCF,4

movf keymatrix+7,W

btfsc IOCCF,5

movf keymatrix+8,W

btfsc IOCCF,6

movf keymatrix+9,W

btfsc IOCCF,7

movf keymatrix+0,W

clrf IOCAF

clrf IOCCF

movwf next

retfie

start:

banksel ANSELA

clrf ANSELA ;Banks A & C as digital

clrf ANSELB

clrf ANSELC

banksel ODCONB ;Bank B as open drain output

movlw 0xFF

movwf ODCONB

banksel INLVLA ;Banks A & C ST input

movwf INLVLA

movwf INLVLC

banksel WPUA ;Banks A & C with internal Pull Up

movwf WPUA

movwf WPUC

banksel TRISA ;Banks A & C all inputs

movwf TRISA

movwf TRISC

clrf TRISB ;Bank B all outputs

banksel ZCDCON

bcf ZCDCON,EN ;ZCD interferes on PORTB.0

banksel PMD2

movlw b'00000111'

movwf PMD2 ;Disable ZCD and comparators

banksel OSCCON1

movlw b'01100000' ;Internal oscillator

movwf OSCCON1

movlw b'00000011' ;8MHz

movwf OSCFRQ

banksel PIE0

movlw b'00010000' ;IOC interrupt enable

movwf PIE0

banksel IOCAN

movlw b'11000000' ;PORTA 6-7 negative interrupt

movwf IOCAN

movlw b'11111111'

movwf IOCCN ;PORTC 0-7 negative interrupt

clrf IOCAF

clrf IOCCF

banksel INTCON

clrf stm ;Reset program variables

clrf stat

movlw 0xFF ;Set the initial keyboard status

movwf next

movwf keymatrix ;to unpressed

movwf keymatrix+1

movwf keymatrix+2

movwf keymatrix+3

movwf keymatrix+4

movwf keymatrix+5

movwf keymatrix+6

movwf keymatrix+7

movwf keymatrix+8

movwf keymatrix+9

bsf INTCON,GIE ;Enable global interrupts

loop:

btfss ps2\_clk ;PS/2 clock polling

goto clkneg

bsf prev\_clk ;Copy positive clock

goto loop

clkneg:

btfss prev\_clk

goto loop

clknegedge:

bcf prev\_clk ;Copy negative clock

bcf data\_copy

btfsc ps2\_data

bsf data\_copy ;Save data pin

movlw StartBit

subwf stm,w

btfsc ZERO

goto is\_StartBit

movlw ParityBit

subwf stm,w

btfsc ZERO

goto is\_ParityBit

movlw StopBit

subwf stm,w

btfsc ZERO

goto is\_StopBit

shift\_register:

lsrf char ;Logical right shift

btfss data\_copy

goto do\_shift

bsf char,7 ;PS/2 is least significant bit first

movlw 0xFF

xorwf parity ;If data=1 apply xor mask

do\_shift:

incf stm

goto loop

is\_StartBit:

clrf char ;Clean for a new character

clrf parity

btfsc data\_copy ;Start bit must be low

goto loop ;If it's high, state machine keeps in 0

incf stm ;If it's low, stm increments

goto loop

is\_ParityBit:

clrf WREG ;Apply xor mask with data\_copy

btfsc data\_copy

movlw 0xFF

xorwf parity

btfss parity\_bit ;If parity=1 wait for stop bit

goto parity\_error

incf stm

goto loop

parity\_error:

clrf stm ;If parity=0 means an error

goto loop ;Reset state machine

is\_StopBit:

clrf stm ;In the tenth bit always reset stm

btfss data\_copy ;Stop bit must be high

goto loop

movlw ExtKey ;Here, we have the data verified

subwf char,w ;If it's an extended key, save it in flag

btfss ZERO ;and wait for another char

goto chk\_release

bsf extended

goto loop

chk\_release:

movlw Break

subwf char,w ;If it's a release, save it in flag

btfss ZERO ;and wait for another char

goto chk\_keycode

bsf release

goto loop

chk\_keycode:

btfsc extended ;Some of the required keys (cursors...)

bsf char,7 ;are extended keys, set bit 7 to translate

movlw LOW(keymatrix)

movwf FSR1L ;Load the pointer to the CPC keys matrix

movlw HIGH(keymatrix)

movwf FSR1H

movlw LOW(keycodes) ;Load the pointer to the keycodes table (-1)

movwf FSR0L

movlw HIGH(keycodes)

movwf FSR0H

lslf char,w ;Multiply char by 2 (2 bytes per keycode)

btfsc CARRY ;If the result is greater than 256...

incf FSR0H ;...high pointer increments

addwf FSR0L ;add low pointer

moviw FSR0++ ;Get keycode and increments pointer

btfsc ZERO ;If keycode is zero, ignore key

goto ignore\_key

addlw LOW(keymatrix-1) ;Points to the correct byte in the matrix

movwf FSR1L

moviw 0[FSR0] ;Get the mask to turn on/off

btfss release

goto press\_key ;If release is activated...

iorwf INDF1 ;... turn on key bit

goto ignore\_key

press\_key:

comf WREG ;If it's a press invert mask

andwf INDF1 ;And turn off key bit

ignore\_key:

bcf release

bcf extended ;Finish matrix update

goto loop

org (HIGH($)+1<<8)

keycodes:

;////////////////////////First 256 bytes

dt 0x00,0x00 ;/////////////////00//////////////

dt 0x01,0x04 ;F9 key (CPC key 3)

dt 0x00,0x00

dt 0x02,0x10 ;F5 key (CPC key 12)

dt 0x01,0x20 ;F3 key (CPC key 5)

dt 0x02,0x20 ;F1 key (CPC key 13)

dt 0x02,0x40 ;F2 key (CPC key 14)

dt 0x00,0x00 ;F12 key (not used)

dt 0x00,0x00

dt 0x02,0x80 ;F10 key (CPC F0 key 15)

dt 0x02,0x08 ;F8 key (CPC key 11)

dt 0x01,0x10 ;F6 key (CPC key 4)

dt 0x03,0x10 ;F4 key (CPC key 20)

dt 0x09,0x10 ;TAB key (CPC key 68)

dt 0x03,0x40 ;ºª\ key (CPC \` key 22)

dt 0x00,0x00

dt 0x00,0x00 ;////////////////10///////////////

dt 0x02,0x02 ;Left ALT key (CPC copy key 9)

dt 0x03,0x20 ;Left SHIFT key (CPC both Shift keys 21)

dt 0x00,0x00

dt 0x03,0x80 ;Left CTRL key (CPC key 23)

dt 0x09,0x08 ;qQ key (CPC key 67)

dt 0x09,0x01 ;1!| key (CPC key 64)

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x09,0x80 ;zZ key (CPC key 71)

dt 0x08,0x10 ;sS key (CPC key 60)

dt 0x09,0x20 ;aA key (CPC key 69)

dt 0x08,0x08 ;wW key (CPC key 59)

dt 0x09,0x02 ;2"@ key (CPC key 65)

dt 0x00,0x00

dt 0x00,0x00 ;////////////////20///////////////

dt 0x08,0x40 ;cC key (CPC key 62)

dt 0x08,0x80 ;xX key (CPC key 63)

dt 0x08,0x20 ;dD key (CPC key 61)

dt 0x08,0x04 ;eE? key (CPC key 58)

dt 0x08,0x01 ;4$~ key (CPC key 56)

dt 0x08,0x02 ;3·# key (CPC key 57)

dt 0x00,0x00

dt 0x00,0x00

dt 0x06,0x80 ;SPACE bar key (CPC key 47)

dt 0x07,0x80 ;vV key (CPC key 55)

dt 0x07,0x20 ;fF key (CPC key 53)

dt 0x07,0x08 ;tT key (CPC key 51)

dt 0x07,0x04 ;rR key (CPC key 50)

dt 0x07,0x02 ;5%? key (CPC key 49)

dt 0x00,0x00

dt 0x00,0x00 ;////////////////30///////////////

dt 0x06,0x40 ;nN key (CPC key 46)

dt 0x07,0x40 ;bB key (CPC key 54)

dt 0x06,0x10 ;hH key (CPC key 44)

dt 0x07,0x10 ;gG key (CPC key 52)

dt 0x06,0x08 ;yY key (CPC key 43)

dt 0x07,0x01 ;6&¬ key (CPC key 48)

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x05,0x40 ;mM key (CPC key 38)

dt 0x06,0x20 ;jJ key (CPC key 45)

dt 0x06,0x04 ;uU key (CPC key 42)

dt 0x06,0x02 ;7/ key (CPC key 41)

dt 0x06,0x01 ;8( key (CPC key 40)

dt 0x00,0x00

dt 0x00,0x00 ;////////////////40///////////////

dt 0x05,0x80 ;,; key (CPC ,< key 39)

dt 0x05,0x20 ;kK key (CPC key 37)

dt 0x05,0x08 ;iI key (CPC key 35)

dt 0x05,0x04 ;oO key (CPC key 34)

dt 0x05,0x01 ;0= key (CPC 0\_ key 32)

dt 0x05,0x02 ;9) key (CPC key 33)

dt 0x00,0x00

dt 0x00,0x00

dt 0x04,0x80 ;//.: key (CPC .> key 31)

dt 0x04,0x40 ;//-\_ key (CPC /? key 30)

dt 0x05,0x10 ;lL key (CPC key 36)

dt 0x04,0x20 ;ñÑ key (CPC key 29)

dt 0x04,0x08 ;pP key (CPC key 27)

dt 0x04,0x02 ;'? key (CPC \_= key 25)

dt 0x00,0x00

dt 0x00,0x00 ;/////////////////50///////////////

dt 0x00,0x00

dt 0x04,0x10 ;´¨{ key (CPC ;: key 28)

dt 0x00,0x00

dt 0x04,0x04 ;`^[ key (CPC @| key 26)

dt 0x04,0x01 ;¡¿ key (CPC arrow/currency key 24)

dt 0x00,0x00

dt 0x00,0x00

dt 0x09,0x40 ;SHIFT LOCK key (CPC key 70)

dt 0x03,0x20 ;Right SHIFT key (CPC key 21)

dt 0x03,0x04 ;RETURN key (CPC key 18)

dt 0x03,0x02 ;+\*] key (CPC [\* key 17)

dt 0x00,0x00

dt 0x03,0x08 ;çÇ} key (CPC ]+ key 19)

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00 ;////////////////60///////////////

dt 0x03,0x40 ;<> key (CPC \` key 22)

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x0A,0x80 ;Backspace key (CPC key 79)

dt 0x00,0x00

dt 0x00,0x00

dt 0x02,0x20 ;1 Keypad key (CPC F1 key 13)

dt 0x00,0x00

dt 0x03,0x10 ;4 Keypad key (CPC F4 key 20)

dt 0x02,0x04 ;7 Keypad key (CPC F7 key 10)

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x02,0x80 ;////////////////70/////////////;0 Keypad key (CPC F0 key 15)

dt 0x01,0x80 ;. Keypad key (CPC keypad . key 7)

dt 0x02,0x40 ;2 Keypad key (CPC F2 key 14)

dt 0x02,0x10 ;5 Keypad key (CPC F5 key 12)

dt 0x01,0x10 ;6 Keypad key (CPC F6 key 4)

dt 0x02,0x08 ;8 Keypad key (CPC F8 key 11)

dt 0x09,0x04 ;ESCAPE key (CPC key 66)

dt 0x00,0x00 ;NUM Lock key (not used)

dt 0x00,0x00 ;F11 key (not used)

dt 0x00,0x00 ;+ Keypad key (not used)

dt 0x01,0x20 ;3 Keypad key (CPC F3 key 5)

dt 0x00,0x00 ;- Keypad key (not used)

dt 0x00,0x00 ;\* Keypad key (not used)

dt 0x01,0x08 ;9 Keypad key (CPC F9 key 3)

dt 0x00,0x00 ;DESP LOCK key (not used)

dt 0x00,0x00

dt 0x00,0x00 ;////////////////80///////////////

dt 0x00,0x00

dt 0x00,0x00

dt 0x02,0x04 ;//F7 key (CPC key 10)

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00 ;/////////////////90///////////////

dt 0x00,0x00 ;//AltGr Key (not used)

dt 0x00,0x00

dt 0x00,0x00

dt 0x03,0x80 ;//Right CTRL key (CPC key 23)

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00 ;//Left WIN key (not used)

dt 0x00,0x00 ;/////////////////A0///////////////

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00 ;//Right WIN key (not used)

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00 ;//SUBMENU key (not used)

dt 0x00,0x00 ;/////////////////B0///////////////

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00 ;/////////////////C0///////////////

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00 ;BackSlash Keypad key (not used)

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00 ;/////////////////D0///////////////

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x01,0x40 ;Enter Keypad key (CPC Intro key 6)

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00 ;/////////////////E0///////////////

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00 ;End Key (not used)

dt 0x00,0x00

dt 0x02,0x01 ;Left Arrow key (CPC key 8)

dt 0x00,0x00 ;Home key (not used)

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00 ;/////////////////F0/////////////; Insert key (not used)

dt 0x03,0x01 ;Del Key (CPC key 16)

dt 0x01,0x04 ;Down Arrow key (CPC key 2)

dt 0x00,0x00

dt 0x01,0x02 ;Right Arrow key (CPC key 1)

dt 0x01,0x01 ;Up Arrow key (CPC key 0)

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00 ;AV Pag key (not used)

dt 0x00,0x00

dt 0x00,0x00

dt 0x00,0x00 ;RE Pag key (not used)

dt 0x00,0x00

dt 0x00,0x00

end

; banksel IOCAF

; clrf WREG

; btfsc IOCAF,7

; movf keymatrix,W

; btfsc IOCAF,6

; movf keymatrix+1,W

; btfsc IOCCF,0

; movf keymatrix+2,W

; btfsc IOCCF,1

; movf keymatrix+3,W

; btfsc IOCCF,2

; movf keymatrix+4,W

; btfsc IOCCF,3

; movf keymatrix+5,W

; btfsc IOCCF,4

; movf keymatrix+6,W

; btfsc IOCCF,5

; movf keymatrix+7,W

; btfsc IOCCF,6

; movf keymatrix+8,W

; btfsc IOCCF,7

; movf keymatrix+9,W

; clrf IOCAF

; clrf IOCCF

; banksel PORTB

; movwf PORTB

; retfie