

```

;*****
;Programa que decodifica las teclas de un teclado matricial 4x4 y
;despliega en un display de 7 segmentos el numero hexadecimal co-
;rrespondiente a la tecla pulsada, utilizando un microcontrolador
;pic 16f628a
;Fecha: 13.06.07 Autor: Jorge A. Bojorquez
;http://micropic.wordpress.com
;*****

```

```

list      p=16f628a
include   p16f628a.inc
__config  0x3F18

cblock   0x20
NUM
TECLA
endc

NUM.7    equ    b'11011000' ; 7
NUM.8    equ    b'00000000' ; 8
NUM.9    equ    b'00010000' ; 9
LET.F    equ    b'00001110' ; F
NUM.4    equ    b'00011001' ; 4
NUM.5    equ    b'00010010' ; 5
NUM.6    equ    b'00000010' ; 6
LET.E    equ    b'00000110' ; E
NUM.1    equ    b'11011001' ; 1
NUM.2    equ    b'01000100' ; 2
NUM.3    equ    b'01010000' ; 3
LET.d    equ    b'01000001' ; D
LET.A    equ    b'00001000' ; A
NUM.0    equ    b'10000000' ; 0
LET.b    equ    b'00000011' ; B
LET.C    equ    b'10000110' ; C

org      0x00
goto     INICIO

org      0x05
INICIO   movlw   0x07           ; Deshabilitar comparadores
         movwf   CMCON         ; analogicos
         bsf    STATUS,RP0     ; Cambio a banco 1
         movlw   0xF0         ; Nibble alto del puerto B como
         movwf   TRISE        ; entrada y bajo como salida
         clrf   TRISA         ; Puerto A como salida
         bcf    OPTION_REG,NOT_RBPU ; Resistencias de pull-up
         ; habilitadas
         bcf    STATUS,RP0     ; Cambio a banco 0

         movlw   NUM.0        ; Iniciar Puerto A para que
         movwf   PORTA        ; despliegue un "0"

PRINCIPAL
         call   TECLADO       ; Llamar a rutina de teclado
         movwf PORTA         ; Desplegar numero en puerto A
         goto  PRINCIPAL     ; Ejecucion ciclica del programa

;*****
;** Rutina que escanea un teclado matricial 4x4 **
;** recorriendo un 0 por cada una de sus filas y **
;** leyendo el estado de cada columna, si la co- **
;** lumna se encuentra con un estado logico alto **
;** no se presiono ninguna tecla, si se encuen- **
;** tra en bajo (0) entonces se detecta la tecla **
;** presionda **
;*****
TECLADO  clrf    TECLA        ; Limpiar variable TECLA
         movlw   b'00001110' ; Poner un cero en la primer
         movwf   PORTB        ; fila del puerto B (RB0)

CHECA_COL
         btfs   PORTB,4       ; Si la 1er columna es "0"

```

```

goto    ANTIRREBOTES      ; salta a la rutina ANTIRREBOTES
incf    TECLA,f           ; Si es "1" incrementa TECLA
btfs    PORTB,5           ; Si la 2da columna es "0"
goto    ANTIRREBOTES      ; salta a la rutina ANTIRREBOTES
incf    TECLA,f           ; Si es "1" incrementa TECLA
btfs    PORTB,6           ; Si la 3er columna es "0"
goto    ANTIRREBOTES      ; salta a la rutina ANTIRREBOTES
incf    TECLA,f           ; Si es "1" incrementa TECLA
btfs    PORTB,7           ; Si la 4ta columna es "0"
goto    ANTIRREBOTES      ; salta a la rutina ANTIRREBOTES
incf    TECLA,f           ; Si es "1" incrementa TECLA

; Si no se detecto ninguna pulsacion se realiza una comparacion
; entre la variable TECLA y el numero "16", si TECLA es menor que
; 16 el "0" en las filas del puerto B se recorre hacia la izquierda
; hacia la siguiente fila, si TECLA es igual a "16" la rutina del
; TECLADO vuelve a comenzar
    movlw    d'16'
    subwf    TECLA,w
    btfsc    STATUS,Z
    goto    TECLADO
    bsf     STATUS,C
    rlf     PORTB,f
    goto    CHECA_COL

;*****
;** Rutina que elimina los rebotes y ademas decodifica **
;** la tecla pulsada y regresa el valor binario neces- **
;** para desplegar los numeros de 0 a F en un display **
;** de 7 segmentos conectado al puerto A **
;*****
ANTIRREBOTES
B1    btfs    PORTB,4
      goto    B1
B2    btfs    PORTB,5
      goto    B2
B3    btfs    PORTB,6
      goto    B3
B4    btfs    PORTB,7
      goto    B4

    movf    TECLA,w
    call    DECOD_TECLA
    return

DECOD_TECLA
    addwf    PCL,f
    retlw    NUM.7
    retlw    NUM.8
    retlw    NUM.9
    retlw    LET.F
    retlw    NUM.4
    retlw    NUM.5
    retlw    NUM.6
    retlw    LET.E
    retlw    NUM.1
    retlw    NUM.2
    retlw    NUM.3
    retlw    LET.d
    retlw    LET.A
    retlw    NUM.0
    retlw    LET.b
    retlw    LET.C

end

```