

```

1  ;-----;
2  ; Dal Degan Santiago - 45421137
3  ; Ejercicio 2 - Punto 2
4  ; En este programa se encienden los leds conectados a RB0, RB1, RB2, RB3 espera un segundo y los
   apaga
5  ;-----;
6
7  ; se configura el pic
8  #include <p16f628a.inc>
9      LIST      P=16f628a
10
11      org 0
12
13      ; configuramos los puertos
14      bsf STATUS, RP0 ; cambiamos al segundo banco de memoria
15      movlw b'11110000'
16      movwf TRISB ; ponemos desde RB0 hasta RB3
17      bcf STATUS, RP0 ; volvemos al primer banco
18
19  INICIO
20      call ENCENDER_POR_SEGUNDO ; Encender los leds cada un segundo
21      goto INICIO ; repetimos
22
23  ENCENDER_TODOS
24      movlw b'00001111' ; ponemos el 1 en los ultimos 4 bits de PORTB
25      movwf PORTB
26      return
27
28  ENCENDER_POR_SEGUNDO
29      call ENCENDER_TODOS ; Encendemos todos los leds
30      call DELAY1s ; delay de un segundo
31      clrf PORTB ; apalagamos los leds
32      call DELAY1s ; delay de un segundo
33      clrwdt ; limpiamos el watchdog
34      return
35
36  DELAY1s
37      ; estamos andando a 4Mhz
38      ; un ciclo de instruccion son 4 ciclos de relojs es decir 4/4 = 1Mhz
39      ; para calcular el tiempo hacemos 1/1Mhz = 1us
40      ; si queremos lograr un delay de 1s necesitamos
41      ; 1.000.000 ciclos de maquina
42      ; sin embargo como toma 3 ciclos de maquina hacer el proceso
43      ; dividimos 1.000.000/3 = 333.333,333...
44      ; ya que no entra eso en un registro lo separaremos en 3
45      ; por cada valor del un registro el otro registro contara
46      ; regresivamente su valor
47      ; es decir reg1=10 reg=20, por cada 10 ciclos restando reg1
48      ; se restara uno de reg2
49      ; para saber los valores necesitamos reg1*reg2*reg3 = 333.333
50      ; raiz cubica 333.333 = 69.3
51
52      movlw d'69'
53      movwf 0x20
54  REG2
55      movlw d'69'
56      movwf 0x21
57  REG3
58      movlw d'70'
59      movwf 0x22
60
61  START
62      decfsz 0x22, 1
63      goto START
64      decfsz 0x21, 1
65      goto REG3
66      decfsz 0x20, 1
67      goto REG2
68      return
69
70  end

```