



18. – Ejercicios

Introducción a los microprocesadores
2015

Ex Feb/2012. Pr2 (modificado)

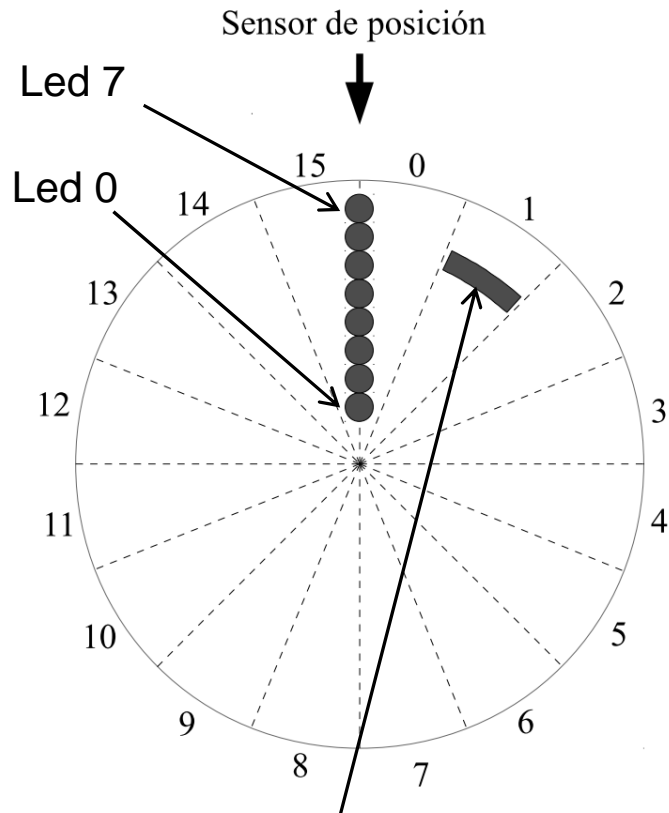


Imagen con led 6 prendido en el sector 1.

“0”: led apagado

“1”: led prendido

Velocidad nominal de giro: 40ms/vuelta

- El disco gira muy rápido y los leds prenden/apagan en la frontera de cada sector → Imagen
- Sensor de posición (**sens_pos**): da un pulso a 0 al pasar el disco por el comienzo del sector 0.
- Estimar la velocidad de cada vuelta utilizando **sens_pos** y así, encender/apagar los leds.
- Datos de la imagen en una tabla de 16 bytes en RAM a partir de la dirección *tbl_led*.
- Solución por interrupciones pues programa ppal se encuentra realizando otras tareas.
- Utilizar un timer (timer1) para medir el tiempo de una vuelta y otro (timer2) para interrumpir al comienzo de cada sector. Utilizar la mayor precisión posible.
- Además **sens_pos** deberá interrumpir con mayor prioridad para determinar la velocidad y actualizar timers.

Ex Feb/2012. Pr2 (modificado)

Se pide:

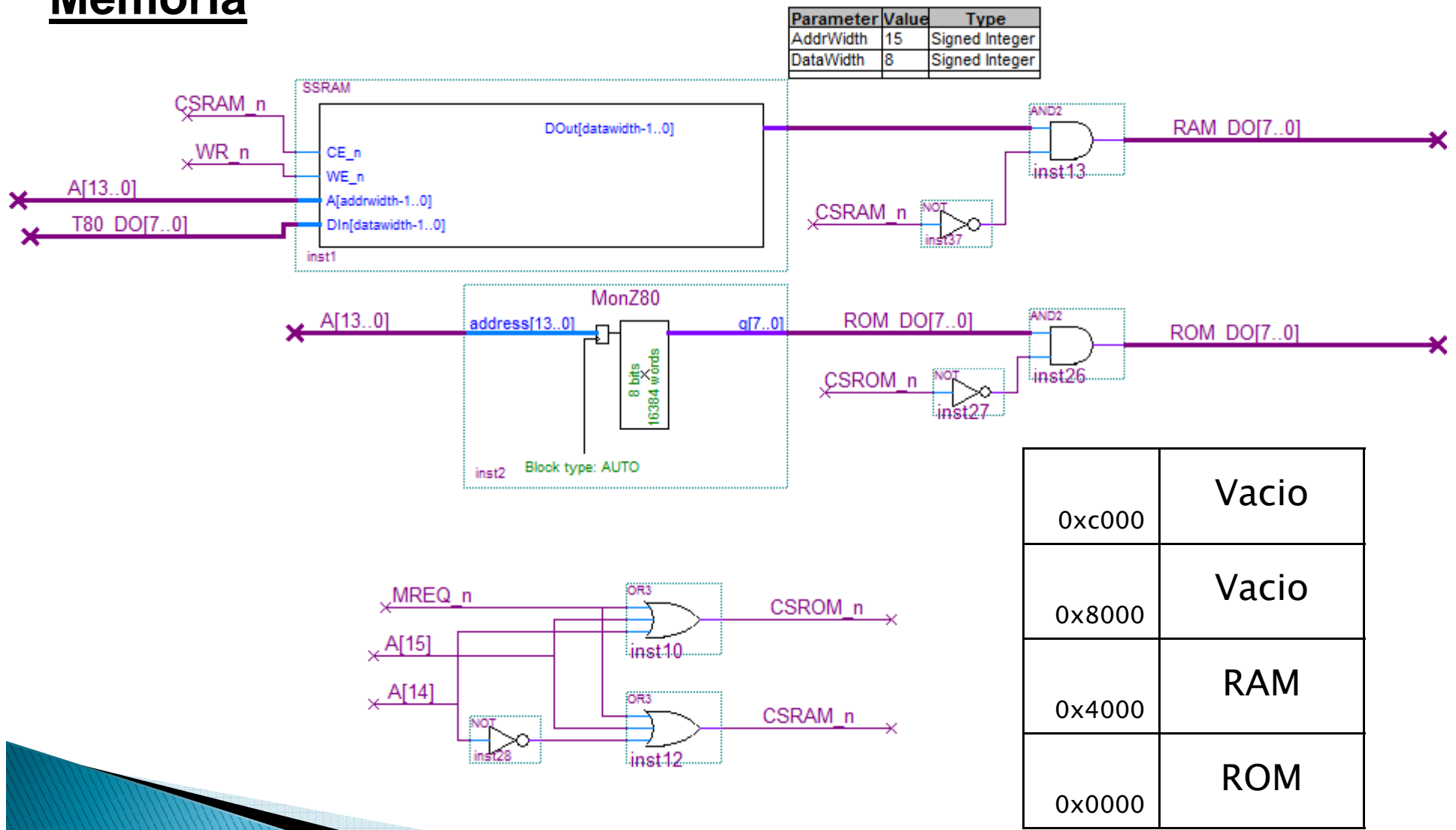
1. Hardware completo del sistema con lógica multiplexada:
 - T80 a 1MHz, 2 Timers y 2 Ctrls Int.
 - Memoria: 16K ROM, 16K RAM en direcciones más bajas. El resto está reservado para futuras aplicaciones.
 - Otros puertos de entrada: **p_in0** y **p_in1** de 8 bits
 - Otros puertos de salida: **p_out0** y **p_out1** de 8 bits
2. Rutinas de atención a las interrupciones:
 - a) Pulso **sens_pos**: debe medir el tiempo de una vuelta y configurar los timers. Además, si la velocidad del disco es tal que se desborda la cuenta del tiempo de la vuelta, se deben apagar los leds durante toda la siguiente vuelta.
 - b) Timer de tiempo de sector para realizar el encendido/apagado de los leds.
3. Inicialización y reservas de memoria

Nota: Considerar que no hay mucha variación de velocidad entre 2 vueltas consecutivas.

Ex Feb/2012. Pr2 (modificado)

Solución:

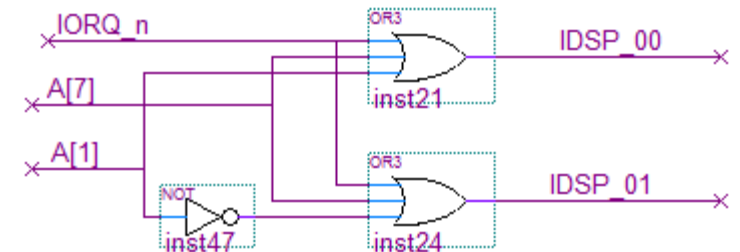
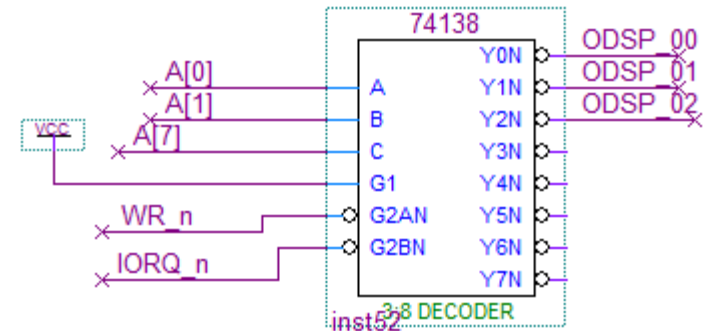
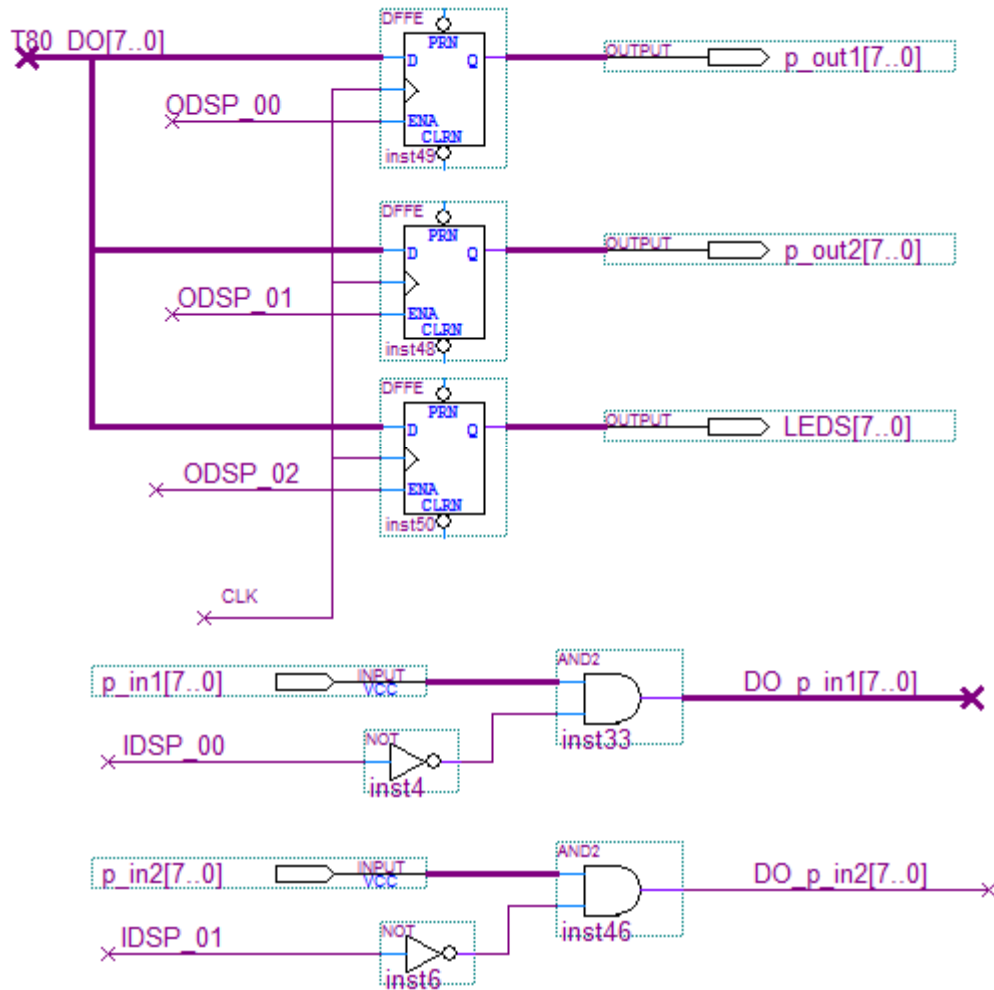
Memoria



Ex Feb/2012. Pr2 (modificado)

Solución:

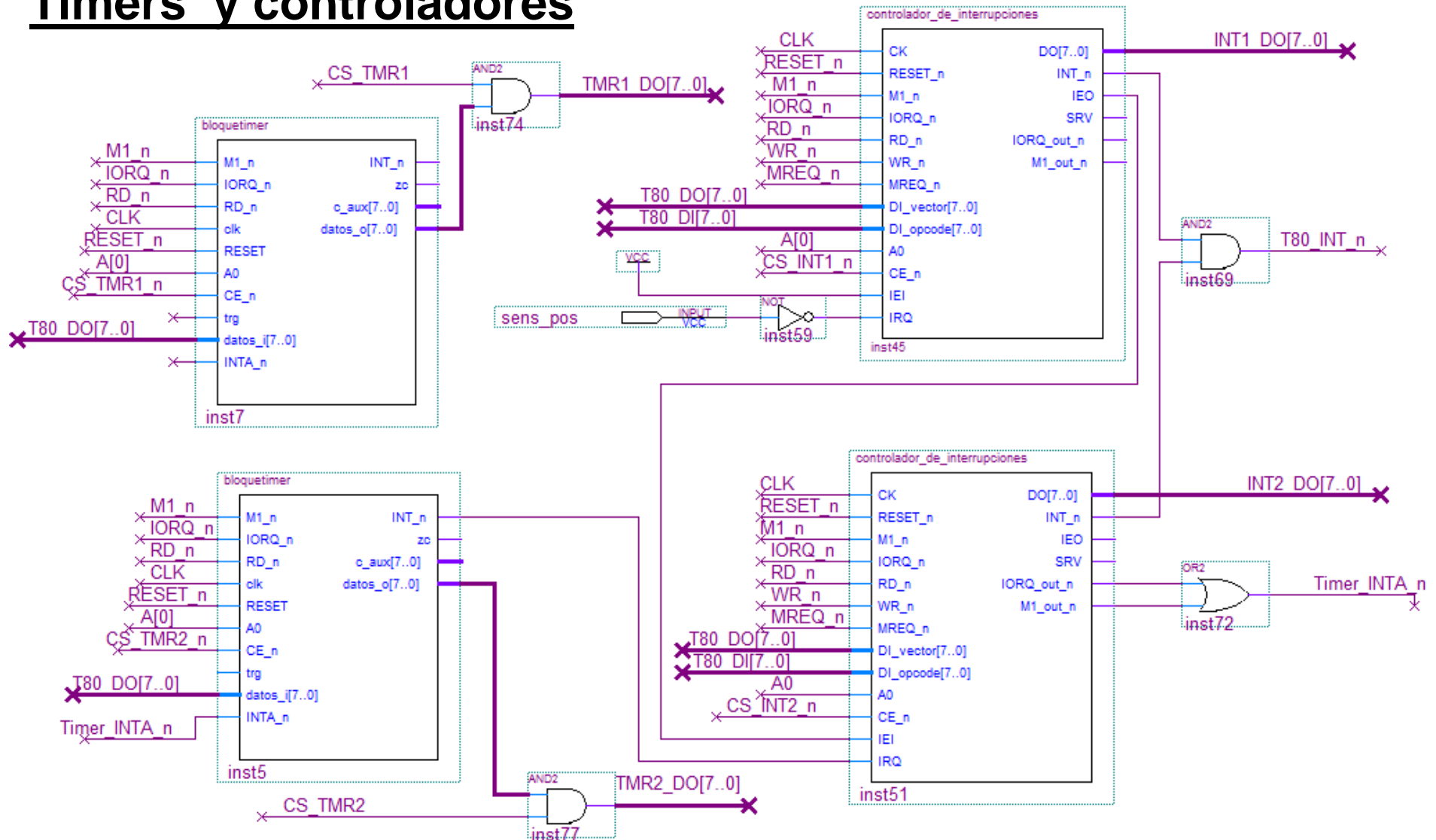
Puertos



Ex Feb/2012. Pr2 (modificado)

Solución:

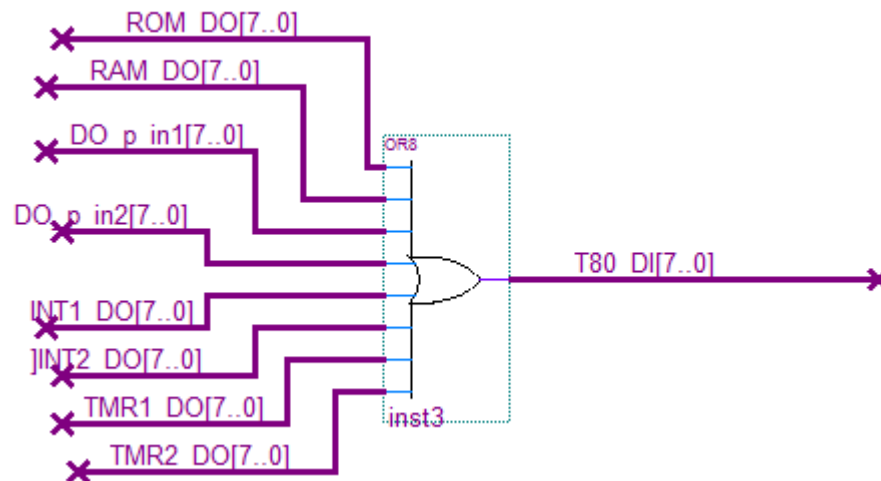
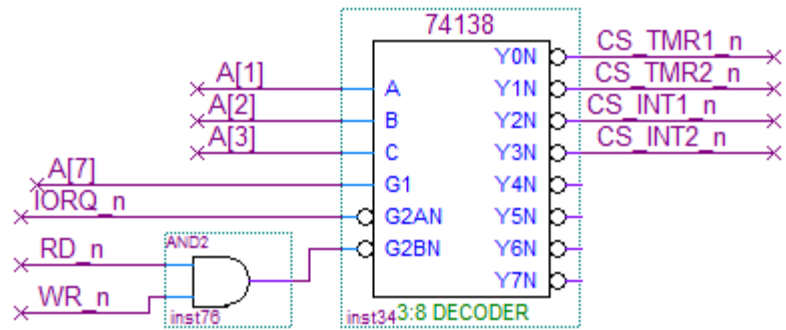
Timers y controladores



Ex Feb/2012. Pr2 (modificado)

Solución:

Timers y controladores (deco)



Ex Feb/2012. Pr2 (modificado)

Solución:

```
TMR1_CTE      equ 80h
TMR1_CTRL     equ 81h
TMR1_ZC       equ 80h
TMR1_Cuenta   equ 81h
TMR2_CTE      equ 82h
TMR2_CTRL     equ 83h
TMR2_ZC       equ 82h
TMR2_Cuenta   equ 83h
```

```
INT1_Vector   equ 84
INT1_CLR_FF   equ 85
INT1_Estado   equ 85
INT2_Vector   equ 86
INT2_CLR_FF   equ 87
INT2_Estado   equ 87
```

```
LEDS          equ 02h
```

Dir	Escritura	Lectura
00	P_out1	P_in1
01	P_out2	P_in2
02	Leds	
80	TMR1 CTE	TMR1 ZC
81	TMR1 CTRL	TMR1 Cuenta
82	TMR2 CTE	TMR2 ZC
83	TMR2 CTRL	TMR2 Cuenta
84	INT1 Vector	INT1 Vector
85	INT1 CLR FF	INT1 Estado
86	INT2 Vector	INT2 Vector
87	INT2 CLR FF	INT2 Estado

Ex Feb/2012. Pr2 (modificado)

Solución:

- Timer 1:
 - No interrumpe
 - Flanco de trig : no se va a usar
 - Reset SW
 - Arranque automático
 - $0.1\mu s \times 2^{n-1} \times 256 < 40ms < 0.1\mu s \times 2^n \times 256$; donde n es el prescalar
 $40ms / (0.1\mu s \times 256) = 1562.5$
 $2^{n-1} < 1562.5 < 2^n \rightarrow \text{prescalar} = 11 = 1011b$

```
TMR1_CW      equ      00101011b
```

- Timer 2:
 - Interrupciones habilitadas/deshabilitadas
 - Flanco de trig: no se va a usar
 - Reset SW
 - Arranque automático
 - Debe ir 16 veces más rápido que el Timer 1 \rightarrow prescalar = 7 = 0111b

```
TMR2_CW_EI   equ      10100111b
```

```
TMR2_CW_DI   equ      00100111b
```

Ex Feb/2012. Pr2 (modificado)

Solución:

```
ORG 2000h ; algún lugar en ROM
;-----
ISR_SENSOR_POS:
    ei
    push af
    in a, (TMR1_ZC)
    bit 0,a
    jr nz, ultimo_giro_lento
ultimo_giro_ok:
    in a,(TMR1_Cuenta)
    neg a
    out (TMR2_CTE), a
    ld a, TMR2_CW_EI ; cargo Palabra de control en TMR2
    out (TMR2_CTRL), a
    ld a, 0
    ld (num_sector),a
    Call PrendoLeds
    jr fin
ultimo_giro_lento:
    .....
```

Ex Feb/2012. Pr2 (modificado)

Solución:

```
.....
ultimo_giro_lento:
    ld a, TMR2_CW_DI           ; deshabilito int de sectores
    out (TMR2_CTRL), a
    ld a, 0
    out(Leds), a              ; apago leds
fin:
    ld a, TMR1_CW              ; reprogramo timer 1
    out (TMR1_CTRL), a
    in a, (TMR1_ZC)            ; borro bandera ZC por si se activó
                                ; antes de reprogramar el timer 1.
    pop af
    reti
;-----
isr_sector:
    ei
    Call PrendoLeds
    reti
```

Ex Feb/2012. Pr2 (modificado)

Solución:

;-----

PrendoLeds:

```
    puch af
    puch hl
    push bc
    ld a,(num_sector)
    ld b, 0
    ld c, a
    inc a
    and 0Fh
    ld (num_sector),a
    ld hl,tbl_led
    add hl,bc
    ld a,(hl)
    out (LEDS),a
    pop bc
    pop hl
    pop af
    ret
```

;-----

Ex Feb/2012. Pr2 (modificado)

Solución:

Inicialización

```
                org 4000h          ; comienzo ram
Sector_num db
                org 3000h          ; medio de rom
tabla_int:
                dw isr_sensor_pos:  ; INT1: Sensor de posición
                dw isr_sector       ; INT2: TMR2

                org 0000h
;inicializo modo 2
                ld sp, 8000h        ;fin ram + 1
                ld i, tabla_int / 256
                im2
; inicializo variables
                ld a,0
                ld (num_sector),a
                -----
```

Ex Feb/2012. Pr2 (modificado)

Solución:

```
-----  
;vector de interrupciones  
  ld a,0  
  out (INT1_Vector), a ; vector de int en 0  
  ld a,2  
  out (INT2_vector), a ; vector de int en 2  
  
  in a, (TMR1_ZC) ; borro Zc del timer 1.  
  ld a, 255  
  ld (TMR1_CTE),a  
  ei  
  jp ppal ; salta al prog ppal con otras tareas.
```