

Enunciado:

Se va a implementar un sistema basado en 8085 para el control de un cruce con entre dos calles de una sola dirección (una principal con bastante tráfico y una secundaria con poco tráfico) regulada por semáforos, con las siguientes características:

- La vía principal tiene un semáforo, que estará habitualmente en verde para los coches.
- La vía secundaria tiene un semáforo que estará habitualmente en rojo para los coches.
- La vía principal está atravesada por un paso de peatones.
- Cuando un peatón quiere cruzar, accionará un pulsador, pasados 60s de la pulsación el semáforo de la vía principal pasará a ámbar, y pasados 10s más, pasará a rojo, pasando el de la vía secundaria a verde. Permanecerán así durante 60s, transcurridos los cuales, el semáforo de la vía secundaria pasará a ámbar y 10s después pasará a rojo, a la vez que el semáforo de la vía principal pasa a verde.
- Cuando un coche se para delante del semáforo en la vía secundaria, un sensor lo detectará y se activará. Cuando el sensor se active, ocurrirá una secuencia igual que la descrita anteriormente, pero con otra temporización: inmediatamente, el semáforo principal se pone en ámbar, a los 10s pasan a SP=R SS=V, en otros 10s pasan a SP=R SS=A, en otros 10s pasan a SP=V SS=R.

Solución 1.

Solución usando un 8255 para las entradas y salidas, un 8254 para la temporización y realizando la entrada/salida por polling.

- Mapa de memoria

Se van a mapear todos los dispositivos en memoria, y seguidos, de la siguiente forma:

ROM:

posición base: 0000h.

Posiciones ocupadas: 0000 00xxxx xxxx xxxx

Última posición: 03FFh

Decodificación: $CS = IOM + A_{15} + A_{14} + A_{13} + A_{12} + A_{10}$

La ROM 2716 es de 2KB, pero se está usando como una memoria de 1K (sólo se usan 10 líneas de dirección)

RAM:

posición base: 0400h.

Posiciones ocupadas: 0000 01xxxx xxxx xxxx

Última posición: 07FFh

Decodificación: $CS = IOM + A_{15} + A_{14} + A_{13} + A_{12} + A_{11} + /A_{10}$

La RAM 2016 es de 2KB, pero se está usando como una memoria de 1K (sólo se usan 10 líneas de dirección)

8255:

posición base: 0800h.

Posiciones ocupadas: 0000 1000 0000 00xx

Última posición: 0803h

Decodificación: $CS = IOM + A_{15} + A_{14} + A_{13} + A_{12} + /A_{11} + A_{10} + A_9 + A_8 + A_7 + A_6 + A_5 + A_4 + A_3$

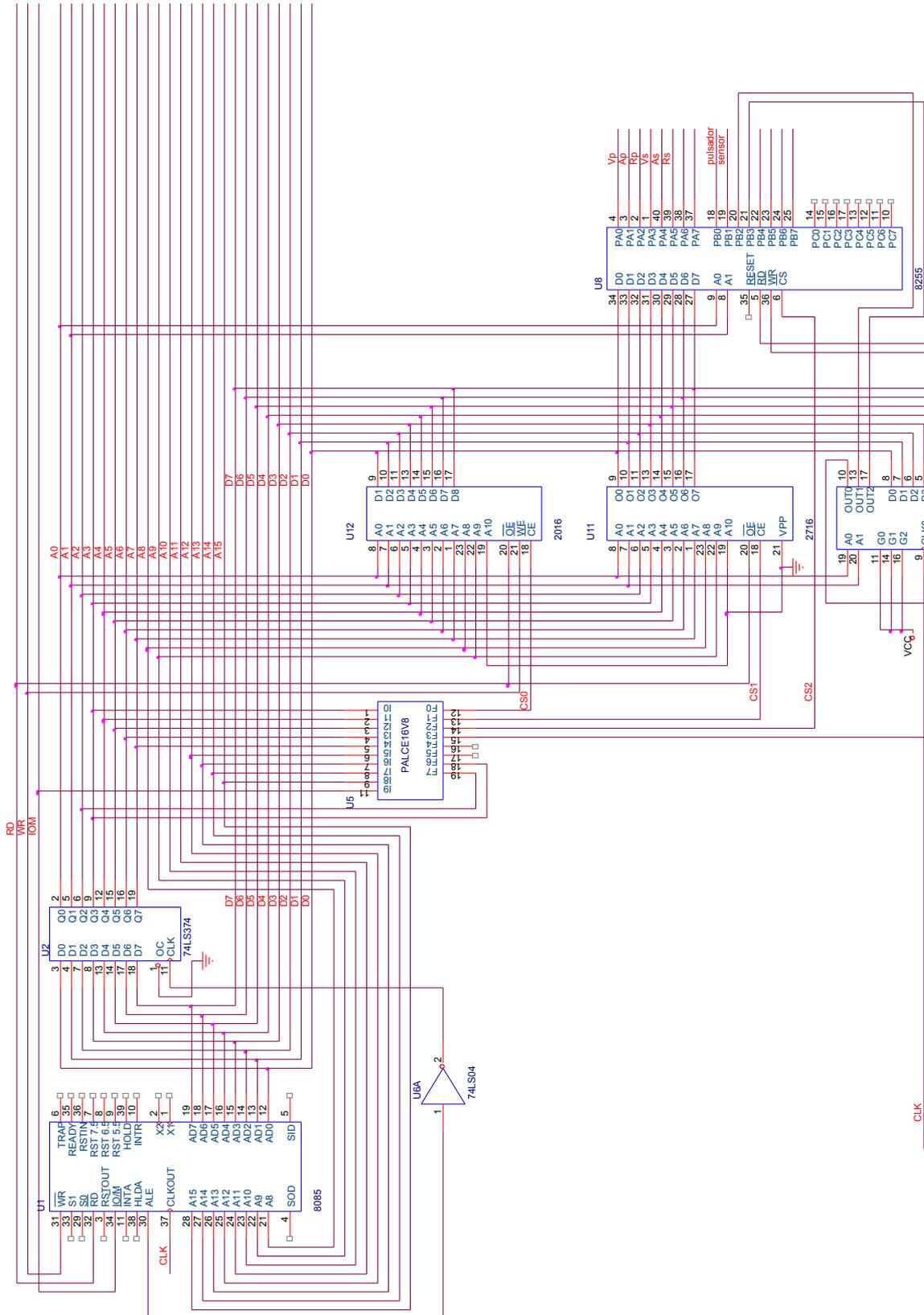
8254:

posición base: 0804h.

Posiciones ocupadas: 0000 1000 0000 01xx

Última posición: 0807h

Decodificación: $CS = IOM + A_{15} + A_{14} + A_{13} + A_{12} + /A_{11} + A_{10} + A_9 + A_8 + A_7 + A_6 + A_5 + A_4 + /A_3$



Title		Logica Digital
Sheet	Document Number	Sistema 8085 sin PIC
B	Rev	1
Date:	Monday, May 19, 2003	Sheet 1 of 3

Configuración de los dispositivos

Usaremos el puerto A del 8255 como salida para controlar las luces de los semáforos y PB como entrada para leer el sensor y el pulsador y las salidas de los temporizadores de 10s y 60s (para controlar por polling cuando terminan de contar).

Programación:

Palabra de control: 82h a la posición de memoria 0803h (registro de control)

Elegimos el cristal del microprocesador, de 2Mhz, de forma que el reloj interno (y el que sale al exterior por el pin CLKOUT) será de 1MHZ.

Programando el temporizador T0 del 8254 en modo divisor de frecuencia (modo 2), con una cuenta 10.000, en binario, la salida OUT0 tiene un periodo de 10ms.

Programación:

Palabra de control: 34h a la posición de memoria 0807h (registro de control)

Inicialización de la cuenta: enviar dos bytes consecutivos: 01h y 27h a la dirección de memoria 0804h (temporizador T0)

El temporizador T1 se programará en modo interrupción a fin de cuenta (modo 0), en BCD, con una cuenta 6.000, lo que dará un tiempo de cuenta de 60s.

Programación:

Palabra de control: 61h a la posición de memoria 0807h (registro de control)

Inicialización de la cuenta: enviar 60h a la dirección de memoria 0805h (temporizador T1)

El temporizador T2 se programará en modo interrupción a fin de cuenta (modo 0), en BCD, con una cuenta 1.000, lo que dará un tiempo de cuenta de 10s.

Programación:

Palabra de control: A1h a la posición de memoria 0807h (registro de control)

Inicialización de la cuenta: enviar 10h a la dirección de memoria 0806h (temporizador T2)

• Programación:

; INICIALIZACION DE LOS DISPOSITIVOS

ORG 0

JMP INICIO ; VECTOR DE RESET. SALTO AL INICIO DEL PROGRAMA.

ORG 50H ; EMPEZAMOS EN LA POSICION 50 PARA SALTAR LAS INTERRUPCIONES

INICIO:

; INICIALIZACION DEL 8255. PUERTO A SALIDA MODO 0,

; PUERTO B ENTRADA MODO 0

MVI A, 82H

STA 0803H

; INICIALIZACION DEL 8254. T0 EN MODO 2, CUENTA 10000

MVI A, 34H

STA 0807H

MVI A, 01H

STA 0804H

MVI A, 27H

STA 0804H

```
;FIJAMOS SEMAFORO PRINCIPAL EN VERDE Y SECUNDARIO EN ROJO
MVI A, 21H
STA 0800H
```

```
;ESPERA A QUE OCURRA UN EVENTO (ACTIVACION DEL SENSOR O PULSACION
; DEL BOTON). ESPERA POR PROGRAMA (POLLING), HACIENDO LECTURA
; CONTINUA DEL PUERTO B Y COMPROBANDO SI CAMBIA ALGUNO DE LOS DOS
```

POLLING:

```
LDA 0801H           ;LEE EL PUERTO B
ANI 00000000B      ;COMPRUEBA SI SE HA PULSADO EL PULSADOR
CZ PULSADOR
```

```
LDA 0801H           ;LEE EL PUERTO B
ANI 00000010B      ;COMPRUEBA SI SE HA ACTIVADO EL SENSOR
CZ SENSOR
```

```
JMP POLLING        ;CONTINUA COMPROBANDO INDEFINIDAMENTE
```

```
;RUTINA DE TEMPORIZACION DEL SEMAFORO CUANDO SE PULSA EL PULSADOR:
;RETARDO DE 60S
;EL SEMAFORO PRINCIPAL PASA A AMBAR
;RETARDO DE 10S
;EL SEMAFORO PRINCIPAL PASA A ROJO Y EL SECUNDARIO A VERDE
;RETARDO DE 60S
;EL SEMAFORO PRINCIPAL PASA A AMBAR
;RETARDO DE 10S
;VUELTA A LA SITUACIÓN INICIAL
```

PULSADOR:

```
;PROGRAMACION DE T1 PARA RETARDO DE 60S
MVI A, 61H
STA 0807H
MVI A, 60H
STA 0805H
```

```
ESPERA A QUE EL TEMPORIZADOR LLEGUE A FIN DE CUENTA
; (LA SALIDA OUT1, CONECTADA A PB2, SE PONE A 1)
ESPERA60:
```

```
           MVI A, 0801H           ;LEE PUERTO B
           ANI 00001000B          ;COMPRUEBA OUT1
JNZ ESPERA60      ;ESPERA HASTA QUE SE PONE A 1
```

```
MVI A, 00100010B      ;PRINCIPAL EN AMBAR
STA 0800H              ;ACTUALIZA EL PUERTO A
```

```
;PROGRAMACION DE T2 PARA RETARDO DE 10S
MVI A, A1H
STA 0807H
MVI A, 10H
STA 0806H
```



```
;RUTINA DE TEMPORIZACION DEL SEMAFORO CUANDO SE ACTIVA EL SENSOR:
;EL SEMAFORO PRINCIPAL PASA A AMBAR
;RETARDO DE 10S
;EL SEMAFORO PRINCIPAL PASA A ROJO Y EL SECUNDARIO A VERDE
;RETARDO DE 10S
;EL SEMAFORO PRINCIPAL PASA A AMBAR
;RETARDO DE 10S
;VUELTA A LA SITUACIÓN INICIAL
```

SENSOR:

```
MVI A, 00100010B           ;PRINCIPAL EN AMBAR
STA 0800H                   ;ACTUALIZA EL PUERTO A
```

```
;PROGRAMACION DE T2 PARA RETARDO DE 10S
MVI A, A1H
STA 0807H
MVI A, 10H
STA 0806H
```

```
;ESPERA A QUE EL TEMPORIZADOR LLEGUE A FIN DE CUENTA
; (LA SALIDA OUT2, CONECTADA A PB3, SE PONE A 1)
ESPERA10_3:
    MVI A, 0801H           ;LEE PUERTO B
    ANI 00001000B         ;COMPRUEBA OUT2
JNZ ESPERA10_3            ;ESPERA HASTA QUE SE PONE A 1
```

```
MVI A, 00001100B           ;PRINCIPAL ROJO, SECUNDARIO VERDE
STA 0800H                   ;ACTUALIZA EL PUERTO A
```

```
;PROGRAMACION DE T2 PARA RETARDO DE 10S
MVI A, A1H
STA 0807H
MVI A, 10H
STA 0806H
```

```
;ESPERA A QUE EL TEMPORIZADOR LLEGUE A FIN DE CUENTA
; (LA SALIDA OUT2, CONECTADA A PB3, SE PONE A 1)
ESPERA10_4:
    MVI A, 0801H           ;LEE PUERTO B
    ANI 00001000B         ;COMPRUEBA OUT2
JNZ ESPERA10_4            ;ESPERA HASTA QUE SE PONE A 1
```

```
MVI A, 00010100B           ;SECUNDARIO EN AMBAR
STA 0800H                   ;ACTUALIZA EL PUERTO A
```

```
;PROGRAMACION DE T2 PARA RETARDO DE 10S
MVI A, A1H
STA 0807H
MVI A, 10H
STA 0806H
```

```

;ESPERA A QUE EL TEMPORIZADOR LLEGUE A FIN DE CUENTA
; (LA SALIDA OUT2, CONECTADA A PB3, SE PONE A 1)
ESPERA10_5:
    MVI A, 0801H           ;LEE PUERTO B
    ANI 00001000B        ;COMPRUEBA OUT2
    JNZ ESPERA10_5       ;ESPERA HASTA QUE SE PONE A 1

    MVI A, 00100001B     ;PRINCIPAL VERDE, SECUNDARIO ROJO
    STA 0800H           ;ACTUALIZA EL PUERTO A

RET                       ;FIN DE LA RUTINA DE TEMPORIZACION

```

Solución 2.

Se añade al circuito anterior un 8259. El control de E/S del sensor y el pulsador se hará por interrupción. El control del fin de cuenta de los temporizadores se seguirá haciendo por polling, ya que el proceso es totalmente secuencial.

- Mapa de memoria

El mapa de memoria no se modifica, excepto que se añade al final el 8259, que ocupa dos posiciones de memoria:

8259:

```

posición base:           0808h.
Posiciones ocupadas:    0000 1000 0000 100x
Última posición:       0809h
Decodificación: CS=IOM+A15+A14+A13+A12+A11+A10+A9+A8+A7+A6+A5+A4+A3+A2

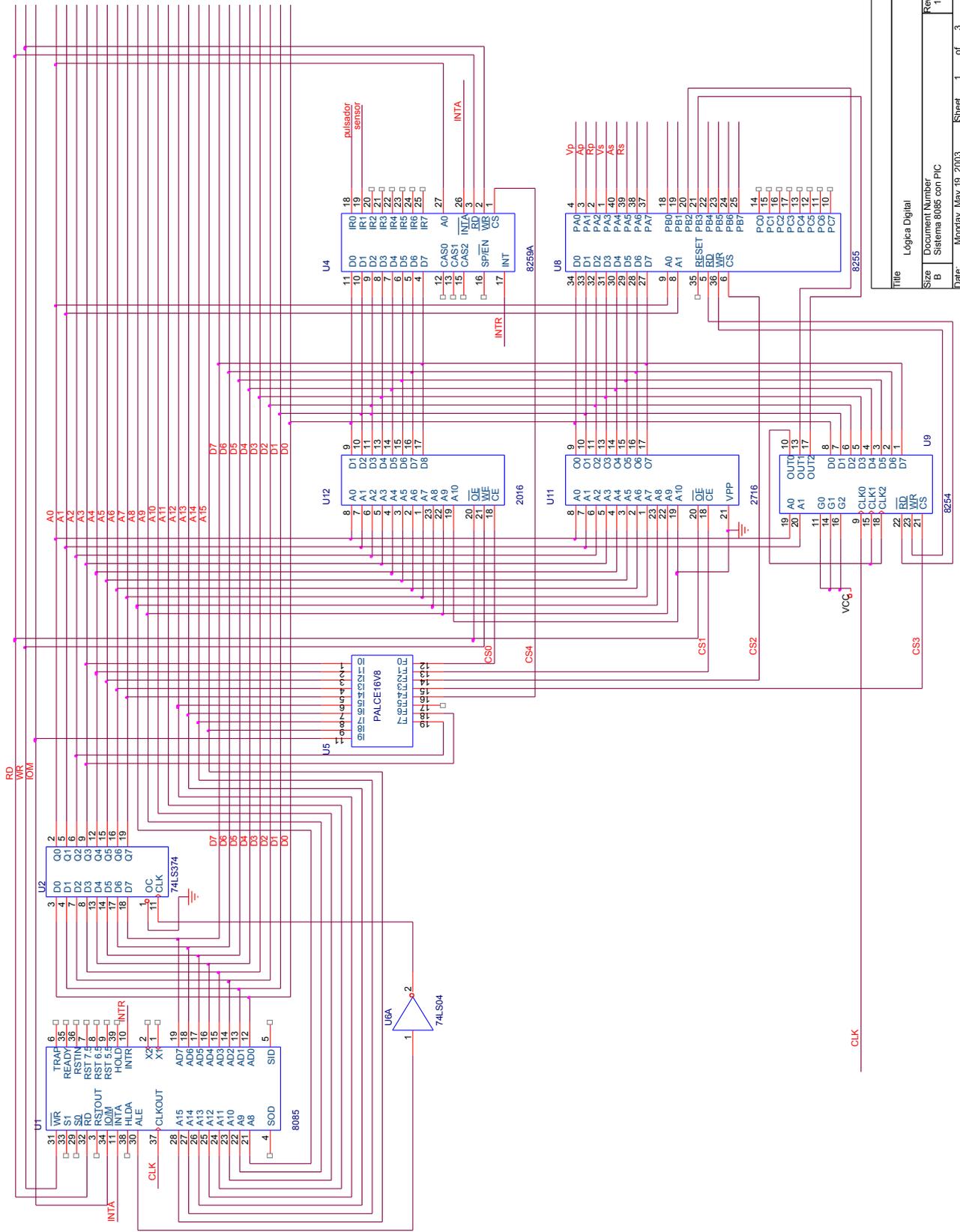
```

- Configuración de los dispositivos

La configuración de los dispositivos será la misma. Además, habrá que configurar el 8259:

Se va a configurar de forma que su dirección base sea 60h (01100000b), con una separación entre vectores de interrupción de 4 bytes, por flanco y modo normal EOI. Por lo tanto, al programarlo para 8085 no será necesaria la ICW4, y al tener un solo 8259, no hace falta tampoco ICW3.

La palabra de inicialización ICW1 será 01110110b, que se escribirá a la posición de memoria 0808h. La palabra de inicialización ICW2 será 00h, que se escribirá a la posición de memoria 0809h.



Title		Logica Digital
Size	Document Number	Sistema 8085 con PIC
B	Rev	1
Date:	Monday, May 19, 2003	Sheet 1 of 3

- Programación:

; INICIALIZACION DE LOS DISPOSITIVOS

ORG 0

JMP INICIO ;VECTOR DE RESET. SALTO AL INICIO DEL PROGRAMA.

ORG 70H

JMP PULSADOR ;VECTORES DE INTERRUPCION.

ORG 74H ;IRQ0 = INTERRUPCION DEL PULSADOR

JMP SENSOR

;IRQ1=INTERRUPCION DEL SENSOR

ORG 70H ;EMPEZAMOS EN LA POSICION 70 PARA SALTAR LAS INTERRUPCIONES

INICIO:

;INICIALIZACION DEL 8255. PUERTO A SALIDA MODO 0,

;PUERTO B ENTRADA MODO 0

MVI A, 82H

STA 0803H

;INICIALIZACION DEL 8254. T0 EN MODO 2, CUENTA 10000

MVI A, 34H

STA 0807H

MVI A, 01H

STA 0804H

MVI A, 27H

STA 0804H

;FIJAMOS SEMAFORO PRINCIPAL EN VERDE Y SECUNDARIO EN ROJO

MVI A, 21H

STA 0800H

;INICIALIZACION DEL 8259

MVI A, 76H ;ICW1

STA 0808H

MVI A, 0H

STA 0809H ;ICW2

;PROGRAMACION DE LA MASCARA DE INTERRUPCION PARA HABILITAR

;SOLO LAS INTERRUPCIONES 0 Y 1

MVI A, 11111100B

STA 0809H

EI

;HABILITAMOS LAS INTERRUPCIONES

;EL PROGRAMA PRINCIPAL NO TIENE NADA QUE HACER YA. ESPERA

;EN UN BUCLE INFINITO

ESPERA: JMP ESPERA

```
;RUTINA DE ATENCION A LA INTERRUPCION DEL PULSADOR
;TEMPORIZACION DEL SEMAFORO CUANDO SE PULSA EL PULSADOR:
;RETARDO DE 60S
;EL SEMAFORO PRINCIPAL PASA A AMBAR
;RETARDO DE 10S
;EL SEMAFORO PRINCIPAL PASA A ROJO Y EL SECUNDARIO A VERDE
;RETARDO DE 60S
;EL SEMAFORO PRINCIPAL PASA A AMBAR
;RETARDO DE 10S
;VUELTA A LA SITUACIÓN INICIAL
```

PULSADOR:

```
;PROHIBIMOS LAS INTERRUCIONE PARA QUE OTRA INTERRUPCION NO
;INTERRUMPA A A ESTA.
```

DI

```
;SE GUARDAN EN LA PILA TODOS LOS REGISTROS
PUSH B
PUSH D
PUSH H
PUS PSW
```

```
;PROGRAMACION DE T1 PARA RETARDO DE 60S
MVI A, 61H
STA 0807H
MVI A, 60H
STA 0805H
```

```
ESPERA A QUE EL TEMPORIZADOR LLEGUE A FIN DE CUENTA
; (LA SALIDA OUT1, CONECTADA A PB2, SE PONE A 1)
```

```
ESPERA60:
                MVI A, 0801H           ;LEE PUERTO B
                ANI 00001000B        ;COMPRUEBA OUT1
JNZ ESPERA60    ;ESPERA HASTA QUE SE PONE A 1
```

```
MVI A, 00100010B           ;PRINCIPAL EN AMBAR
STA 0800H                  ;ACTUALIZA EL PUERTO A
```

```
;PROGRAMACION DE T2 PARA RETARDO DE 10S
MVI A, A1H
STA 0807H
MVI A, 10H
STA 0806H
```

```
;ESPERA A QUE EL TEMPORIZADOR LLEGUE A FIN DE CUENTA
; (LA SALIDA OUT2, CONECTADA A PB3, SE PONE A 1)
```

```
ESPERA10:
                MVI A, 0801H           ;LEE PUERTO B
                ANI 00001000B        ;COMPRUEBA OUT2
JNZ ESPERA10    ;ESPERA HASTA QUE SE PONE A 1
```



```
;HABILITAMOS LAS INTERRUPCIONES.  
EI
```

```
RET                ;FIN DE LA RUTINA DE INTERRUPCION.
```

```
;RUTINA DE ATENCION A LA INTERRUPCION DEL SENSOR  
;TEMPORIZACION DEL SEMAFORO CUANDO SE ACTIVA EL SENSOR:  
;EL SEMAFORO PRINCIPAL PASA A AMBAR  
;RETARDO DE 10S  
;EL SEMAFORO PRINCIPAL PASA A ROJO Y EL SECUNDARIO A VERDE  
;RETARDO DE 10S  
;EL SEMAFORO PRINCIPAL PASA A AMBAR  
;RETARDO DE 10S  
;VUELTA A LA SITUACIÓN INICIAL
```

```
SENSOR:
```

```
;PROHIBIMOS LAS INTERRUPCIONES PARA QUE OTRA INTERRUPCION NO  
;INTERRUMPA A ESTA.
```

```
DI
```

```
;SE GUARDAN EN LA PILA TODOS LOS REGISTROS  
PUSH B  
PUSH D  
PUSH H  
PUSH PSW
```

```
MVI A, 00100010B      ;PRINCIPAL EN AMBAR  
STA 0800H              ;ACTUALIZA EL PUERTO A
```

```
;PROGRAMACION DE T2 PARA RETARDO DE 10S  
MVI A, A1H  
STA 0807H  
MVI A, 10H  
STA 0806H
```

```
;ESPERA A QUE EL TEMPORIZADOR LLEGUE A FIN DE CUENTA  
;(LA SALIDA OUT2, CONECTADA A PB3, SE PONE A 1)
```

```
ESPERA10_3:  
    MVI A, 0801H      ;LEE PUERTO B  
    ANI 00001000B    ;COMPRUEBA OUT2  
JNZ ESPERA10_3      ;ESPERA HASTA QUE SE PONE A 1
```

```
MVI A, 00001100B      ;PRINCIPAL ROJO, SECUNDARIO VERDE  
STA 0800H              ;ACTUALIZA EL PUERTO A
```

```
;PROGRAMACION DE T2 PARA RETARDO DE 10S
MVI A, A1H
STA 0807H
MVI A, 10H
STA 0806H
```

```
;ESPERA A QUE EL TEMPORIZADOR LLEGUE A FIN DE CUENTA
; (LA SALIDA OUT2, CONECTADA A PB3, SE PONE A 1)
ESPERA10_4:
    MVI A, 0801H           ;LEE PUERTO B
    ANI 00001000B        ;COMPRUEBA OUT2
JNZ ESPERA10_4          ;ESPERA HASTA QUE SE PONE A 1
```

```
MVI A, 00010100B        ;SECUNDARIO EN AMBAR
STA 0800H               ;ACTUALIZA EL PUERTO A
```

```
;PROGRAMACION DE T2 PARA RETARDO DE 10S
MVI A, A1H
STA 0807H
MVI A, 10H
STA 0806H
```

```
;ESPERA A QUE EL TEMPORIZADOR LLEGUE A FIN DE CUENTA
; (LA SALIDA OUT2, CONECTADA A PB3, SE PONE A 1)
ESPERA10_5:
    MVI A, 0801H           ;LEE PUERTO B
    ANI 00001000B        ;COMPRUEBA OUT2
JNZ ESPERA10_5          ;ESPERA HASTA QUE SE PONE A 1
```

```
MVI A, 00100001B        ;PRINCIPAL VERDE, SECUNDARIO ROJO
STA 0800H               ;ACTUALIZA EL PUERTO A
```

```
;ENVIAMOS AL 8259 UN COMANDO FIN DE INTERRUCION (EOI)
;NO ESPECIFICO
```

```
MVI A, 20H
STA 0808H
```

```
;RECUPERAMOS DE LA PILA LOS REGISTROS
POP PSW
POP H
POP D
POP B
```

```
;HABILITAMOS LAS INTERRUPCIONES.
EI
```

RET

;FIN DE LA RUTINA DE TEMPORIZACION