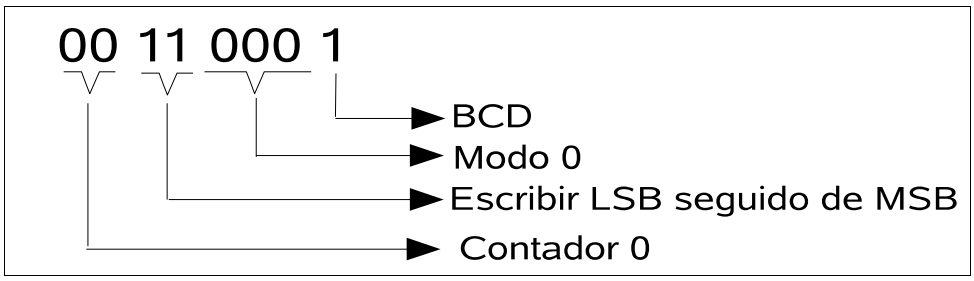
Problema 5

Cálculo de la temporización

Al ser el reloj externo de 100Khz, el periodo es de 10us. La temporización de 0,1s será de 100.000 us, por lo tanto el temporizador deberá contar hasta 10.000.

Programación del 8254

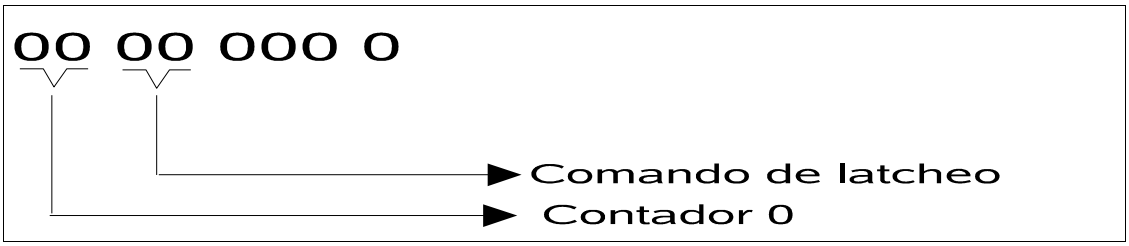
El 8254 se programará inicializándose a 10.000 y se comprobará cuando llega a cero. Se hará por programa, es decir, leyendo la cuenta del temporizador. Po lo tanto, no tiene demasiada importancia el modo que se use, ya que no se va a usar el pin de salida del temporizador para comprobar que llega a fin de cuenta. Voy a programarlo en modo 0, en BCD, inicializándolo a 9.999. La palabra de incialización será:



La palabra de control se escribirá en el registro de control (posición 3 de E/S). A continuación se escribirá el LSB, que es 10011001b (o 99h) y a continuación el MSB, ambos a la posición del TMR0, que es la posición 0 de E/S.

Lectura de la cuenta del temporizador 0

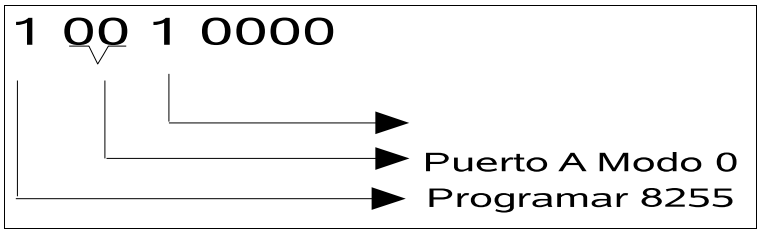
Para comprobar cuando llega a 0 el temporizador 0 habrá que leer su cuenta periódicamente. Para leer la cuenta hay que enviar un comando de latcheo, que es el siguiente:



El comando de latcheo se enviará l registro de control del 8254 (posición 3 de E/S). A continuación se leerá el LSB y luego el MSB ambos de la posición del temporizador 0 (posición 0 de E/S).

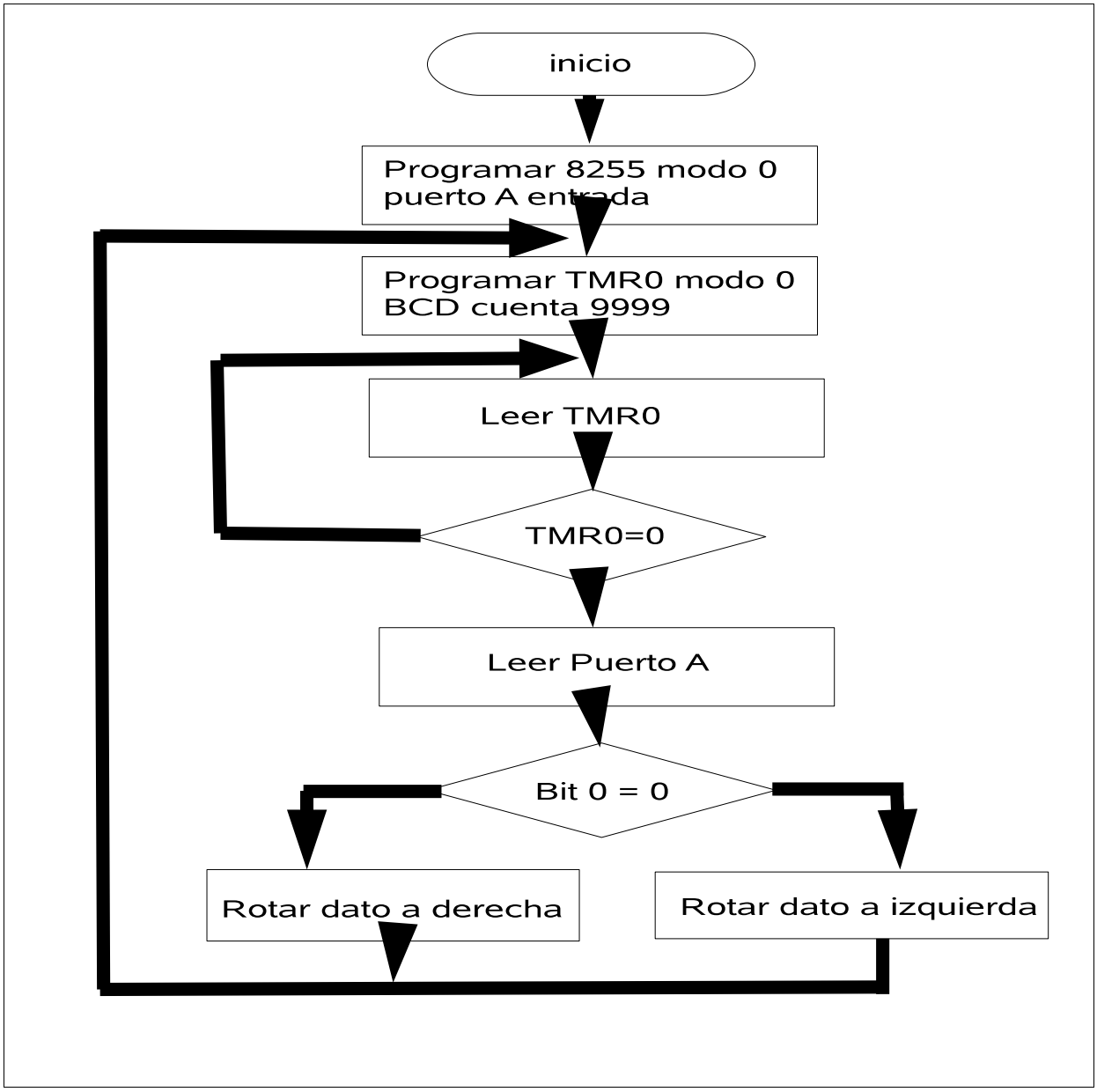
Programación del 8255

El 8255 se programará con el puerto A en modo 0 como entrada. El resto no importa (se escribirán como ceros el resto de bits). La palabra de inicialización será:



La palabra de inicialización se escribirá en el registros de control del 8255 que está en la posición 7 de E/S.

Diagrama de flujo del sistema



Código

```
;--------------------------------------------------
;Defínción de datos
;--------------------------------------------------
DATO  EQU  0100h
;--------------------------------------------------
; Programa principal
;--------------------------------------------------
ORG 0
;programación del 8255, puerto A modo 0
MVI A, 10010000B   ;palabra de control
OUT 7          ;al registro de control

;programación del temporizador 0 modo 0 BCD 9999
PROG:          MVI A, 00110001B   ;palabra de control
               OUT  3             ;al registro de control
               MVI A, 99h
               OUT 0              ;LSB  al temporizador 0
               OUT 0              ;MSB  al temporizador 0
               ;lectura del temporizador 0
               LEER:        MVI A, 0
                            OUT 3         ;comando de latcheo
                            IN 0          ;lectura del LSB
                            CPI 0         ; ¿LSB=0?
                            JNZ LEER      ; espera hasta que LSB=0
                            IN 0          ;lectura del MSB
                            CPI 0         ;¿MSB=0?
                            JNZ LEER      ; espera hasta que MSB=0

               IN 4               ;lectura del puerto A
               ANI 00000001b      ; comprueba el bit 0
               JNZ IZDA           ; si Bit0=1
               DCHA:        LDA  DATO   ;carga el dato
                            RRC         ;rota a la derecha
                            STA DATO    ;lo vuelve a dejar en memoria
                            JMP PROG
```

```
IZDA:   LDA  DATO   ;carga el dato
        RLC         ;rota a la izquierda
        STA DATO    ;lo vuelve a dejar en memoria
        JMP PROG
```

Problema 6

El programa será muy similar, excepto que habrá que inicializar el 8259 y que la parte más importante del programa (la que rota los datos y hace la temporización) no estará en el programa principal sino en la rutina de interrupción, y no se hará con un bucle, sino por interrupción en desbordamiento de la interrupción.

Programación del 8259

Para la programación del 8259 habrá que enviar las palabra de inicialización ICW1 e ICW2 (la ICW3 no es necesaria porque sólo hay un 8259 y la ICW4 no es necesaria porque el micro es un 8085). Se quiere que las posiciones de los vectores de interrupción empiecen en la posición 1000h (0001000000000000b) y estén separados por 4 bytes unos de otros. La interrupción se programará como activa en flanco de subida (no se dice nada en el enunciado sobre esto, se podría elegir también activa por nivel alto). La palabra de inicialización ICW1 será la siguiente:

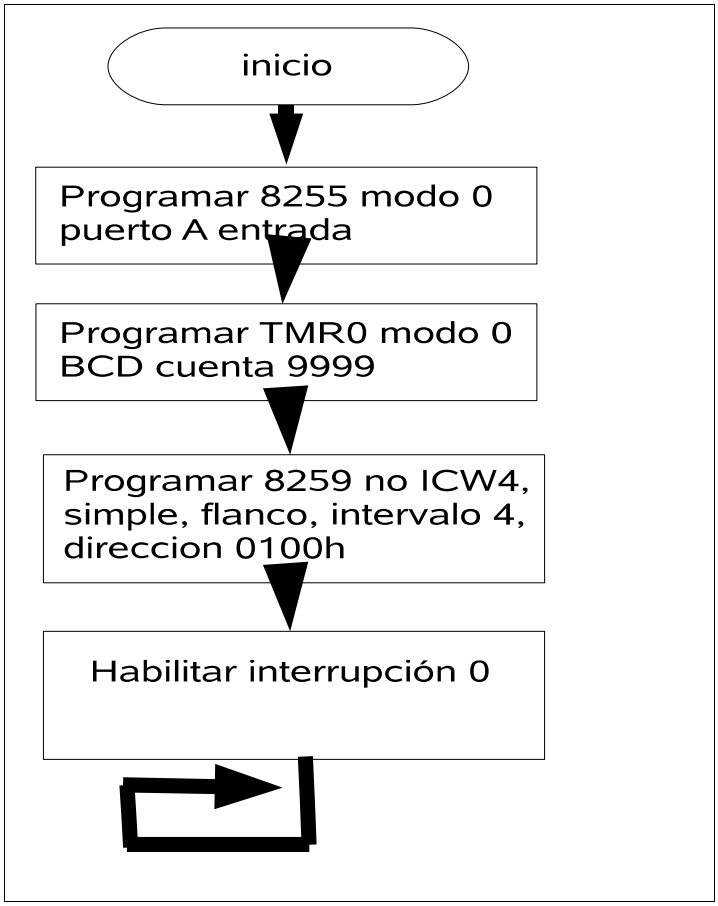
La palabra de inicialización ICW2 será los 8 bits altos de la direccion del inicio de los vectores, es decir, 00010000b

La palabra de operación para habilitar la interrupción 0, será la OCW1, donde cada bit de la palabra de control indica si la interrupción está habilitada (0) o deshabilitada (1), para habilitar sólo la interrupción 0 se enviará un codigo 11111110b al segundo registro del 8259 (dirección 9).

La palabra de operación para indicar el fin de interrupción será la OCW2, donde para indicar un Fin de Interrupción (EOI) automático, se enviará un carácter 00100000b a la dirección del primer registro del 8259 (posición 8).

Diagrama de flujo

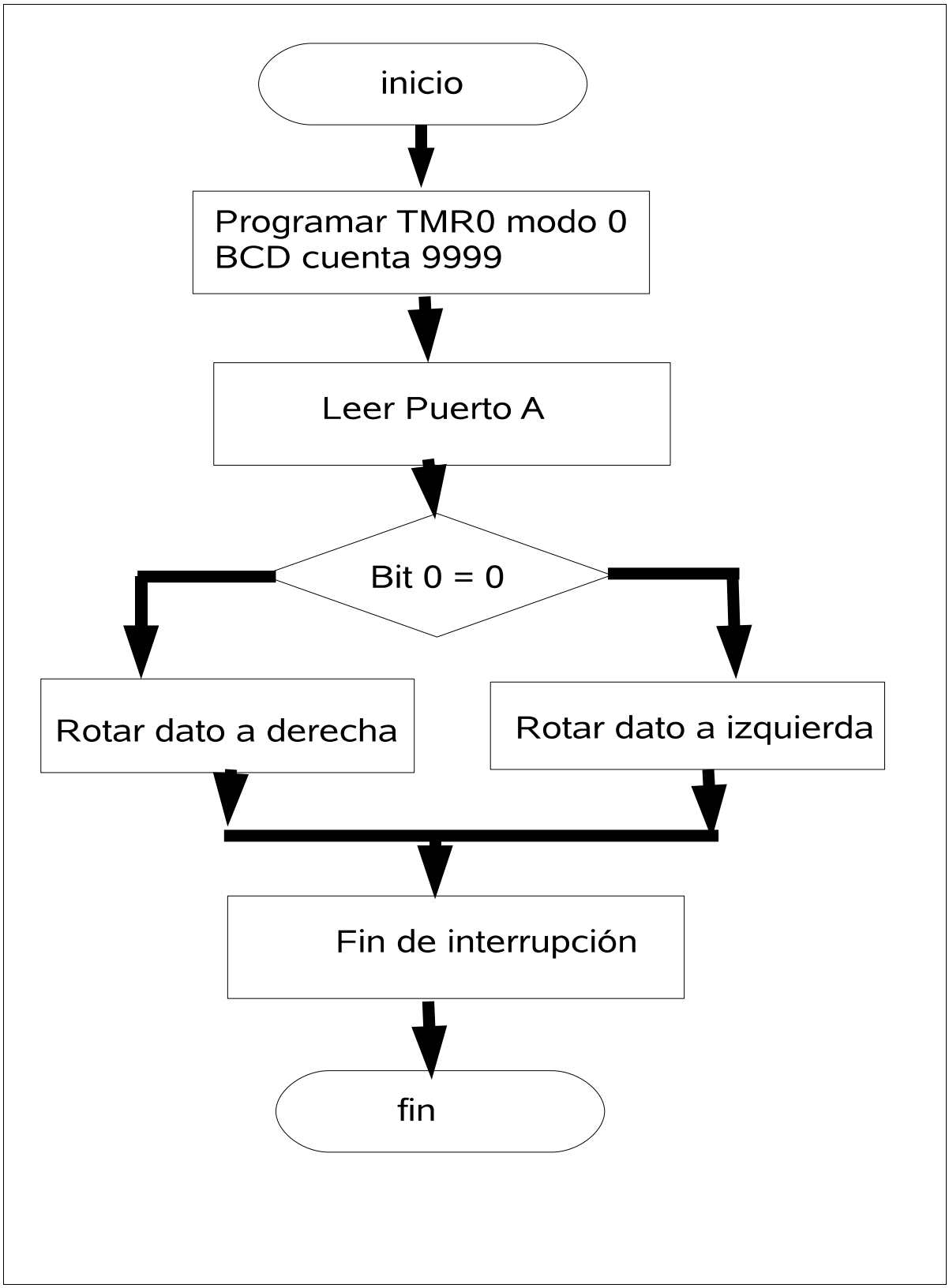
El diagrama de flujo del programa principal será el siguiente



El programa programará los dispositivos, lanzará el temporizador, habilitará las interrupciones y se quedará sin hacer nada en un bucle infinito.

El resto lo hará la interrupción. Cuando el temporizador llegue a fin de cuenta (0), producirá la interrupción 0, que volverá a programar el TMR0 para que vuelva a realizar la temporización, leerá el puerto y rotará el dato.

La rutina de la interrupción, como es la 0, estará en la primera posición de los vectores de interrupción: la 1000h.


inicio
Programar TMR0 modo 0
BCD cuenta 9999
Leer Puerto A
Bit 0 = 0
Rotar dato a derecha
Rotar dato a izquierda
Fin de interrupción
fin

Código

```
;-----------------------------------------------------
;Definición de datos
;-----------------------------------------------------
DATO  EQU  0100h
;-----------------------------------------------------
; Programa principal
;-----------------------------------------------------
ORG 0
;programación del 8255, puerto A modo 0
MVI A, 10010000B   ;palabra de control
OUT 7              ;al registro de control

;programación del temporizador 0 modo 0 BCD 9999
MVI A, 00110001B   ;palabra de control
OUT  3             ;al registro de control
MVI A, 99h
OUT 0              ;LSB  al temporizador 0
OUT 0              ;MSB  al temporizador 0

;programación del 8259
MVI A, 00010110B   ;palabra de inicialización ICW1
OUT 8              ;al primer registro del 8259
MVI A, 00010000b   ;palabra de inicializacion ICW2
OUT 9              ;al segundo registro del 8259

;habilitación de la interrupción 0
MVI A,  11111110b  ;máscara de interrupción
OUT 9              ;habilita la IRQ0

;bucle infinito
BUCLE: JMP BUCLE
```

```
;-------------------------------------------------------
; Rutina de atención a la interrupción 0
;-------------------------------------------------------
ORG 1000h
        PUSH PSW            ;guarda el acumulador y los flags
        ;programación del temporizador 0 modo 0 BCD 9999
        MVI A, 00110001B    ;palabra de control
        OUT  3              ;al registro de control
        MVI A, 99h
        OUT 0               ;LSB  al temporizador 0
        OUT 0               ;MSB  al temporizador 0

        IN 4                ;lectura del puerto A
        ANI 00000001b       ; comprueba el bit 0
        JNZ IZDA            ; si Bit0=1
DCHA:   LDA  DATO   ;carga el dato
        RRC         ;rota a la derecha
        STA DATO    ;lo vuelve a dejar en memoria
        JMP FIN

IZDA:   LDA  DATO   ;carga el dato
        RLC         ;rota a la izquierda
        STA DATO    ;lo vuelve a dejar en memoria

FIN:    MVI A, 00100000b    ;código de fin de interrupción
        OUT 8               ;al 8259
        POP PSW             ;recupera el acumulador y los flags
        RET                 ;fin de la interrupción.
```

Problema 7

Conexiones en el circuito

El 74373 se utiliza para la captura de la parte baja del bus de direcciones. La señal ALE capturará en el registro las direcciones A[7..0].

Todos los dispositivos (8255, 8254, RAM, etc.) se conectan tanto al bus de datos, como al bus de direcciones, como a las señales RD y WR. Además, para cada uno de los dispositivos, se genera un Chip Select a partir de las líneas de dirección (no se usan todas, sino a[15..5] por lo que algunos dispositivos aparecerán repetidos varias veces en el mapa de memoria o E/S) y de la señal IO/M. En el caso de la ROM, como no tiene una señal RD para activar la lectura, pero tiene dos CS, uso uno de los dos CS como señal de lectura (para que se lea la memoria debe estar CS activado y activarse RD). En el caso de la RAM, uso como señal para activar la lectura la señal OE (ya que tampoco tiene señal RD).

El 8254 se ha conectado usando dos temporizadores en cascada, la salida del TMR1 está conectada como entrada de reloj del TMR0. Esto se ha hecho así porque con un temporizador (de 16 bits) sólo se puede hacer una temporización contando hasta 65536. Como el reloj del sistema es de 1MHz (1us) y la temporización que hay que hacer es de 1 minuto, la temporización total (número de pulsos de reloj que hay que contar) sería de 60.000.000, por lo que no vale un solo temporizador. Se usarán dos en cascada de forma que uno temporice 10.000 y el otro 6.000. Se ha usado la conexión desde la salida OUT1 a la entrada CLK0, pero también podría haberse hecho con OUT1 a GATE0 y dejar CLK0 conectada al reloj del sistema.

Generación del mapa de memoria

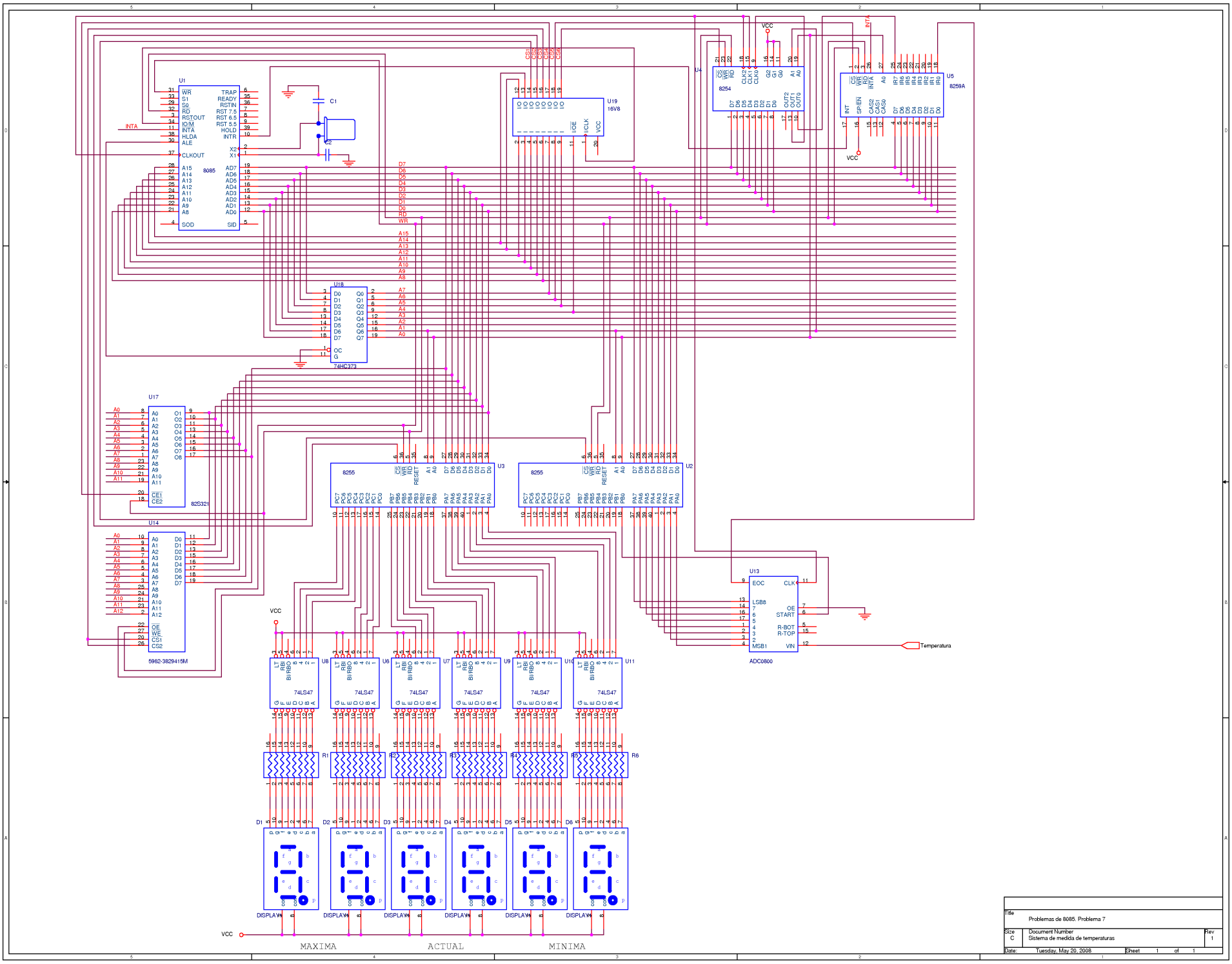
El mapa de memoria que se pide en el enunciado se realiza generando los chip select de los dispositivos con la PAL. Tenemos que CS1 controla el segundo 8255, CS2 el primer 8255 (el de los displays), CS3 controla RAM, CS4 la ROM, CS5 el 8259 y CS6 el 8254. Cada uno de ellos deberá activarse (a nivel bajo) sólo cuando el dispositivo esté direccionado (cuando la dirección en el bus A[15..0] coincida con una de las direcciones de su rango. Como la PAL no tiene entradas suficientes para tener en cuenta A[15..0], se tienen en cuenta sólo A[14..4], lo que quiere decir que los dispositivos estarán repetidos en el mapa de memoria (la RAM y la ROM dos veces, los 8255 ocho veces, el 8254 4 veces y el 8259 ocho veces, como luego se detallará).

La funciones CS1..CS6 son la siguientes

$$CS1 = MIO + \overline{A_{15}} + \overline{A_{13}} + \overline{A_{12}} + \overline{A_{11}} + \overline{A_{10}} + \overline{A_9} + \overline{A_8} + A_7 + A_6 + A_5 + A_4$$
$$CS2 = MIO + \overline{A_{14}} + \overline{A_{13}} + \overline{A_{12}} + \overline{A_{11}} + \overline{A_{10}} + \overline{A_9} + \overline{A_8} + \overline{A_7} + \overline{A_6} + \overline{A_5} + \overline{A_4}$$
$$CS3 = MIO + A_{14} + \overline{A_{13}}$$
$$CS4 = MIO + A_{14} + A_{13} + A_{12}$$
$$CS5 = \overline{MIO} + A_7 + A_6 + A_5 + \overline{A_4}$$
$$CS6 = \overline{MIO} + A_7 + A_6 + A_5 + A_4$$

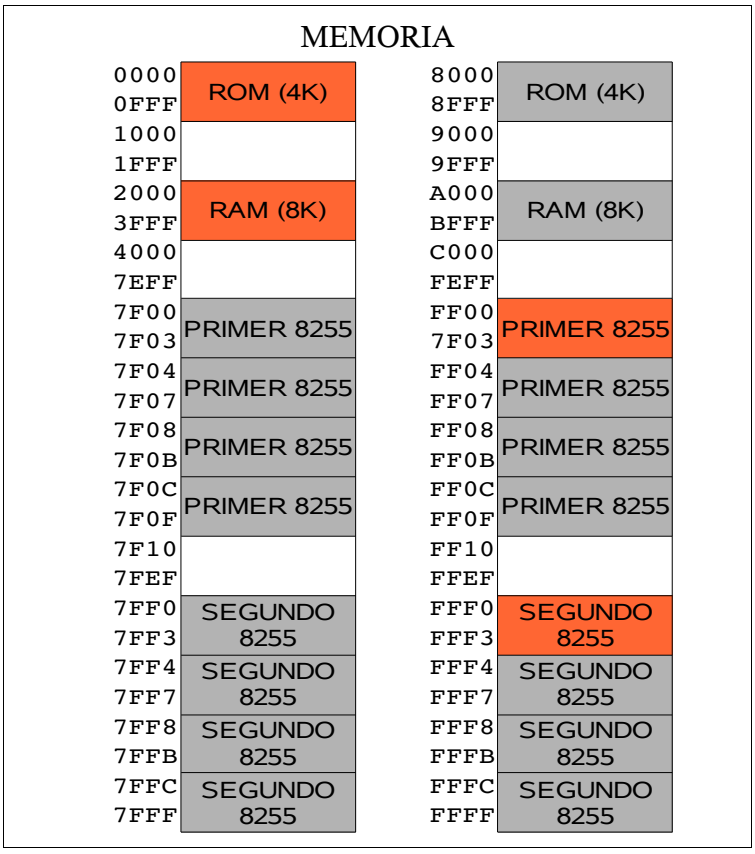
Veamos con estas señales como queda el mapa de memoria:

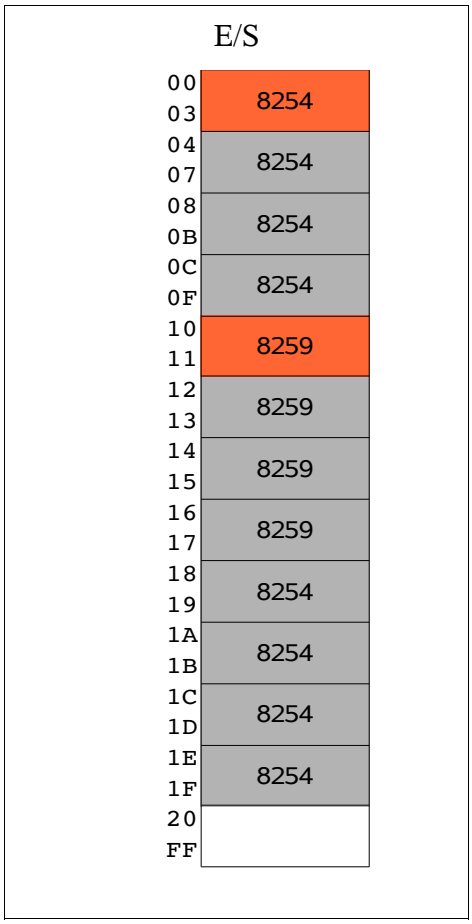
- La ROM, de 4KB, (activada por CS4), queda seleccionada (CS4=0), cuando M/IO está a 0, y A_{14}, A_{13} y A_{12} están también a 0. Estará por lo tanto en el mapa de memoria (M/IO=0) y e las posiciones X000 XXXX XXXX XXXX. Al no usarse A_{15} para la codificación de direcciones, la ROM estará repetida dos veces, una en la posición 0000h y otra en la posición 8000h.


U1
8085
C1
C2
INTA
U19
16V8
U4
8254
U5
8259A
VCC
U18
74HC373
U17
82S321
U14
5962-3829415M
U3
8255
U2
8255
U13
ADC0800
Temperatura
U8
U6
U7
U9
U10
U11
74LS47
R1
R2
R3
R4
R5
R6
DISPLAY¥¥
MAXIMA
ACTUAL
MINIMA
Problemas de 8085. Problema 7
Sistema de medida de temperaturas
Tuesday, May 20, 2008

- La RAM, de 8KB, (activada por CS3) queda seleccionada cuando M/IO está a 0, y A_{14} está a 0 y A_{13} está a 1. Estará por lo tanto en el mapa de memoria (M/IO=0) y e las posiciones X01X XXXX XXXX XXXX. Al no usarse A_{15} para la codificación de direcciones, la RAM estará repetida dos veces, una en la posición 2000h y otra en la posición A000h.
- El 8259, (activado por CS5) queda seleccionada cuando M/IO está a 1, y A_7, A_6, A_5, y A_4 están a 0001. Estará por lo tanto en el mapa de E/S (M/IO=1) y e las posiciones 0001 XXXX (el mapa de E/S no es de 65536 posiciones como el de memoria, sino sólo de 256 posiciones, es decir, sólo se usan A_7.A_0 para las direcciones de E/S). El 8259 ocupa dos posiciones de memoria (tiene como línea de direcciones A_0). Al no usarse A_1, A_2 y A_3 para la codificación de direcciones, el 8259 estará repetida ocho veces veces, en las posiciones 10h, 12h, 14h, 16h, 18h, 1Ah, 1Ch y 1Eh.
- El 8254, (activado por CS6) queda seleccionada cuando M/IO está a 1, y A_7, A_6, A_5, y A_4 están a 0000. Estará por lo tanto en el mapa de E/S (M/IO=1) y e las posiciones 0000 XXXX. El 8254 ocupa 4 posiciones de memoria. Al no usarse A_2 y A_3 para la codificación de direcciones, el 8254 estará repetida cuatro veces, en las posiciones 00h, 04h, 08h y 0Ch.
- Los 8255 (seleccionados por CS1 y CS2), aunque sean dispositivos de E/S, está situados en el mapa de memoria (M/IO=0), en las posiciones X111 1111 0000 XXXX para el segundo 8255 (el de CS1) y X111 1111 1111 XXXX para el primero (CS2). Su posición base será por lo tanto 07F00h para el primer 8255 y 07FF0h para el segundo, que podemos observar que no son las que se piden en el enunciado. Sin embargo, al no usarse para la generación del Chip Select ni A_{15} ni A_3 ni A_2, y tener estos dispositivos 4 posiciones de memoria (líneas de direcciones A_1 y A_0), estarán repetidos 8 veces cada uno: el primer 8255 (el de CS2) estará en 7F00h, 7F04h, 7F08h, 7F0Ch, FF00h, FF04h, FF08h y FF0Ch y el segundo (el de CS1) en 7FF0h, 7FF4h, 7FF8h, 7FFCh, FFF0h, FFF4h, FFF8h y FFFCh. Podemos observar entonces, que si que está cada uno donde se pide en el enunciado, aparte de en otras 7 posiciones de memoria más.

Visto en forma gráfica, el mapa de memoria y de E/S quedará así:





donde en el mapa de memoria se especifican para cada bloque sus direcciones de principio y fin, los bloques en blanco son posiciones de memoria o E/S que no están ocupadas por ningún dispositivo, los bloques en rojo son las posiciones base principales para cada dispositivo (las que se especifican en el enunciado), que son las que nos interesan para usar los dispositivos, y los bloques en gris son posiciones donde los distintos dispositivos aparecen repetidos.

Para generar estos Chip Select se usará una PAL16V8, que irá programada (en ABEL) según el siguiente código:

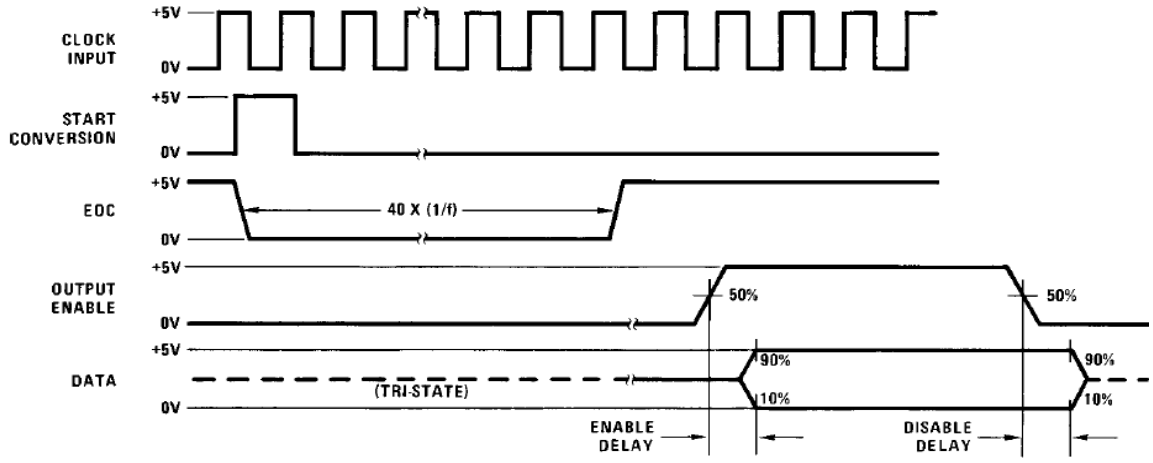
```
MODULE decodificador;
CS1..CS6 PIN 14..19 ISTYPE 'COM';
A14,A13,A4,MIO PIN 12,13,11,1 ISTYPE 'COM';
A12..A5 PIN 2..9 ISTYPE 'COM';

EQUATIONS
CS1=MIO+!A14+!A13+!A12+!A11+!A10+!A9+!A8+A7+A6+A5+A4;
CS2=MIO+!A14+!A13+!A12+!A11+!A10+!A9+!A8+!A7+!A6+!A5+!A4;
CS3=MIO+A14+!A13;
CS4=MIO+A14+A13+A12;
CS5=!MIO+A7+A6+A5+!A4;
CS6=!MIO+A7+A6+A5+A4;

END
```

Utilización del ADC

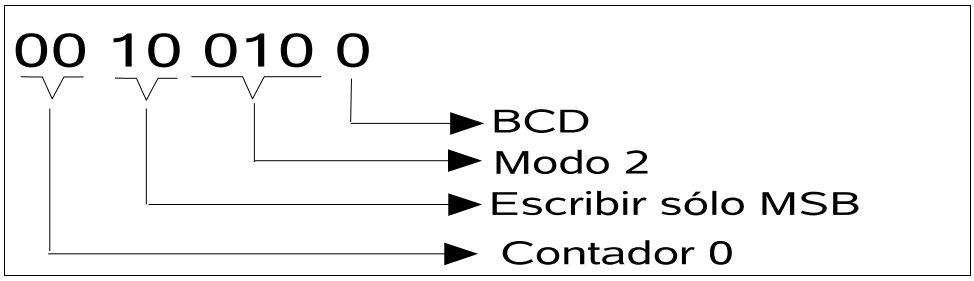
El cronograma para la conversión del ADC es el siguiente:



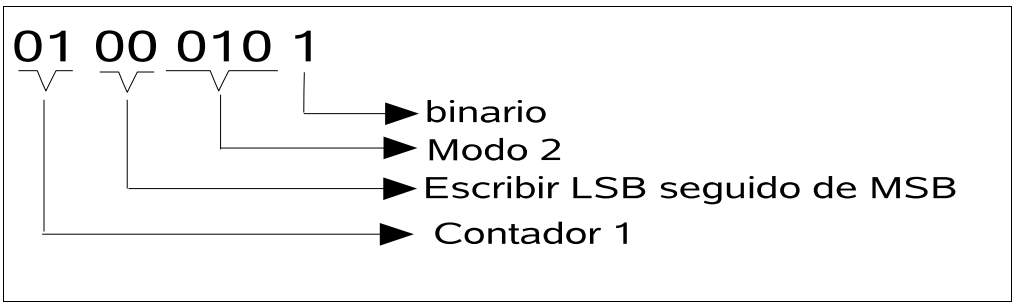
donde vemos que el pulso que hay que dar en START para que el ADC empiece a convertir es a nivel alto (además, mirando en catálogo vemos que debe ser mayor que un periodo de reloj y menor que 3 ½ periodos de reloj. Y que al iniciar la conversión EOC se pone a nivel bajo y al finalizar se pone a nivel alto (lo que producirá un flanco en IRQ0 y por lo tanto que salte la interrupción).

Programación del 8254

Se programarán ambos temporizadores en modo 2 (divisor de frecuencia) y en cascada uno con otro, TMR0 con cuenta 6.000 (en BCD) y TMR1 con cuenta 10.000 (en binario). De esta forma, al estar conectados en cascada hacen en conjunto una temporización de 60.000.000, que con un reloj de 1us, produce un flanco de reloj a la salida de OUT0 cada 60 segundos. Al programarlo en modo divisor de frecuencia no hay que redispararlos, ya que se redisparan solos, por lo que sólo habrá que programarlos al principio del programa (en el programa principal). La palabra de configuración para el TMR0 será:



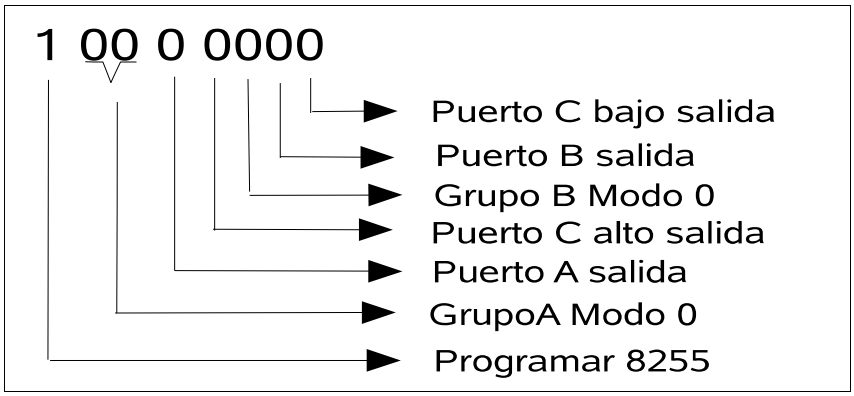
La palabra de control se escribirá en el registro de control (posición 3 de E/S). A continuación se escribirá el MSB, que es 60h a la posición del TMR0, que es la posición 0 de E/S.



La palabra de control se escribirá en el registro de control (posición 3 de E/S). A continuación se escribirá el LSB, que es 00010000b (o 10h) y a continuación el MSB que es 00100111b (27h), ambos a la posición del TMR1, que es la posición 1 de E/S.

Programación del primer 8255

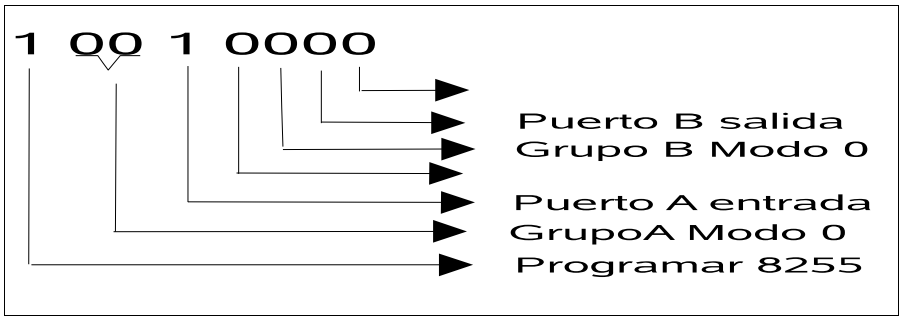
El primer 8255 es el que va conectado a los 6 displays de 7 segmentos. Se programará los tres puertos como salida en modo 0. La palabra de inicialización será:



La palabra de inicialización se escribirá en el registros de control del 8255 que está en la posición FF00h de memoria.

Programación del segundo 8255

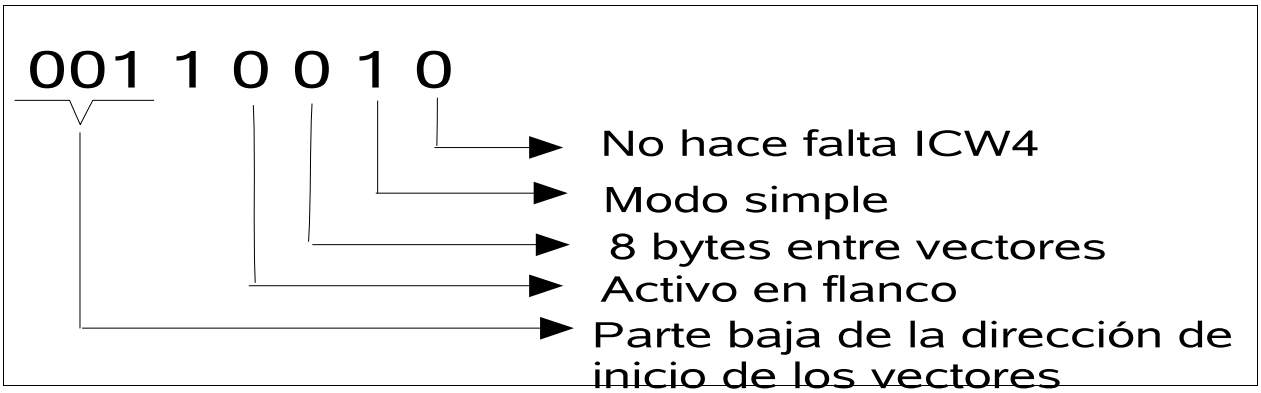
El segundo 8255 va conectado PA como entrada para leer el dato digital del ADC y PB0 como salida para disparar el ADC, por lo tanto Se programará el puerto A como entrada en modo 0 y el puerto B como salida en modo 0.



La palabra de inicialización se escribirá en el registros de control del 8255 que está en la posición FFF0h de memoria.

Programación del 8259

Para la programación del 8259 habrá que enviar las palabra de inicialización ICW1 e ICW2 (la ICW3 no es necesaria porque sólo hay un 8259 y la ICW4 no es necesaria porque el micro es un 8085). No se especifica la posición para los vectores de interrupción, por lo que se tomarán al principio de la memoria (ROM, claro porque es una parte del programa), dejando la posición 0 para el reset (el micro al resetearse empezará a ejecutar código desde la posición 0). La primera posición (distinta de 0) que se puede especificar en la programación del 8255 es con A_5=1, es decir, la posición 0000 0000 0010 0000b (20h). La separación entre interrupciones se hará de 8 bytes. La interrupción se programará como activa en flanco de subida (para que se dispare IRQ0 en el flanco del contador OUT0). La palabra de inicialización ICW1 será la siguiente:



La palabra de incialización ICW1 se enviará al primer registro del 8259, es decir, a la dirección 10h de E/S. La palabra de inicialización ICW2 será los 8 bits altos de la dirección del inicio de los vectores, es decir, 00000000b, que se enviará al segundo registro del 8259 (dirección 11h de E/S).

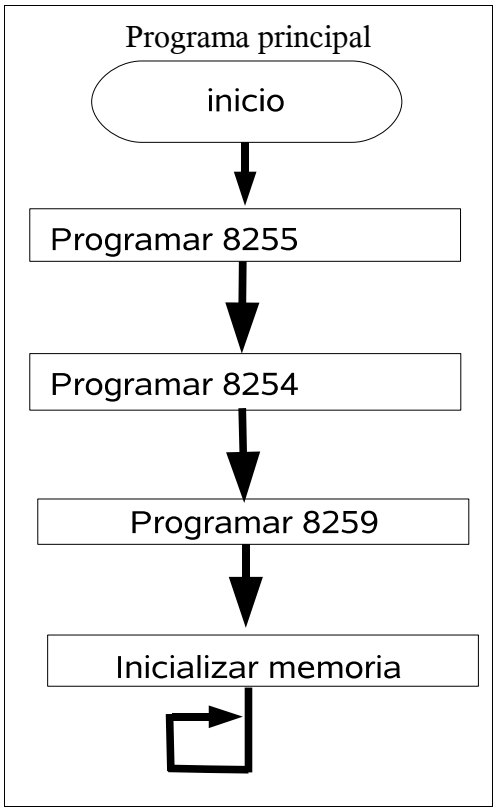
La palabra de operación para habilitar la interrupción 0 y la 7, será la OCW1, donde cada bit de la palabra de control indica si la interrupción está habilitada (0) o deshabilitada (1), para habilitar sólo la interrupción 0 se enviará un codigo 01111110b al segundo registro del 8259..

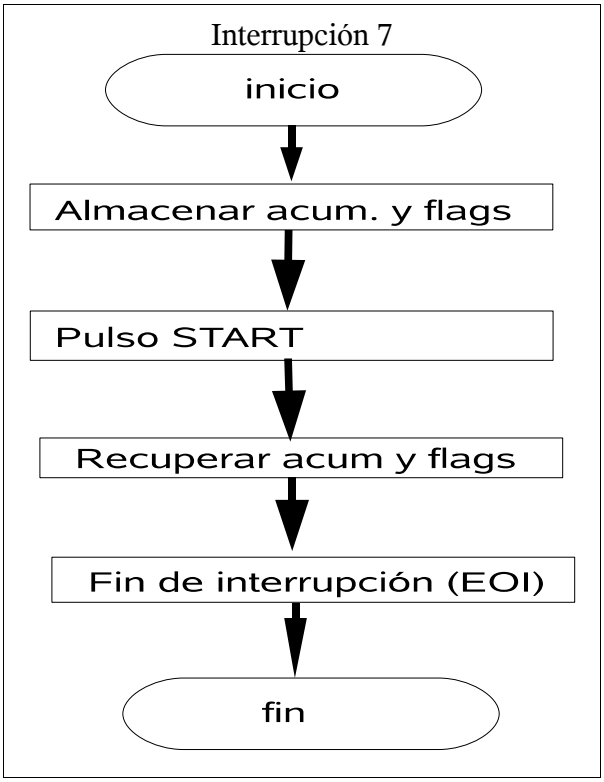
La palabra de operación para indicar el fin de interrupción (tanto de IRQ0 como de IRQ7) será la OCW2, donde para indicar un Fin de Interrupción (EOI) automático, se enviará un carácter 00100000b a la dirección del primer registro del 8259.

Diagrama de flujo

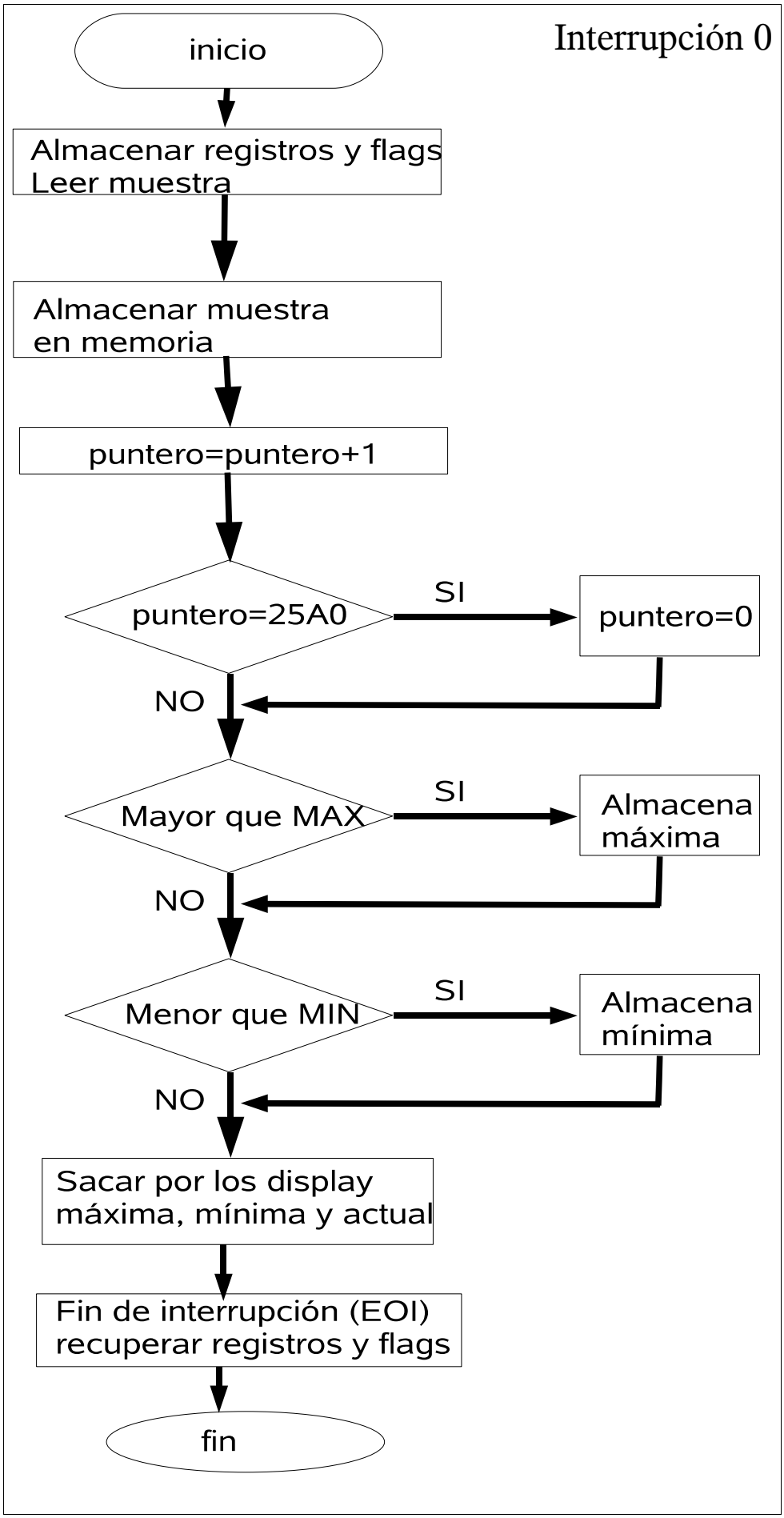
El programa principal simplemente inicializa los dispositivos (y se queda en un bucle infinito)

la rutina de la interrupción 7 (temporizador), lo único que hace es disparar el ADC (en vez de usar una interrupción, se podría conectar por hardware, que sería mejor).





La interrupción 0 será la que hará la mayor parte del trabajo, leyendo las muestras, almacenándolas en memoria (las muestras de hasta un día completo, 60x24=1440 muestras (5A0h), desde la posición de memoria 2100h hasta la 259Fh), almacenando la mayor y la menor y sacando todas ellas por los displays.


Interrupción 0
inicio
Almacenar registros y flags
Leer muestra
Almacenar muestra
en memoria
puntero=puntero+1
puntero=25A0
SI
puntero=0
NO
Mayor que MAX
SI
Almacena
máxima
NO
Menor que MIN
SI
Almacena
mínima
NO
Sacar por los display
máxima, mínima y actual
Fin de interrupción (EOI)
recuperar registros y flags
fin

Código

```
;*********************************************************************
;VARIABLES
;*********************************************************************
;EL PUNTERO OCUPA DOS POSICIONES DE MEMORIA PRIMERO LSB LUEGO MSB
PUNTERO EQU 2000h
MAXIMA EQU 2002h
MINIMA EQU 20003h
ACTUAL EQU 2004h

;*********************************************************************
; VECTORES DE INICIO Y DE LAS INTERRUPCIONES
;*********************************************************************
ORG 0
        JMP PRINCIPAL
ORG 20h
        JMP IRQ0
ORG 58h
        JMP IRQ7

;*********************************************************************
;PROGRAMA PRINCIPAL.
;*********************************************************************
PRINCIPAL:  MVI A, 10000000B   ;PROGRAMACIÓN DEL PRIMER 8255
            STA FF03h
            MVI A,0
            STA  FF00H         ;SACA UN 0 POR EL PUERTO A
            STA  FF01H         ;SACA UN 0 POR EL PUERTO B
            STA  FF01H         ;SACA UN 0 POR EL PUERTO C
            MVI A, 10010000B   ;PROGRAMACIÓN DEL SEGUNDO 8255
            STA FFF3h
            STA  FFF1H         ;PONE A 0 EL PUERTO B
            MVI A, 00100100B   ;PROGRAMACIÓN DEL TMR0
            OUT 3
            MVI A, 60h         ;LSB
            OUT 0
```

```
        MVI A, 01000101B   :PROGRAMACIÓN DEL TMR1
        OUT 3
        MVI A, 10h         ;LSB
        OUT 1
        MVI A, 27h
        OUT 1              ;MSB
        MVI A, 00110010B   ;PROGRAMACIÓN DEL 8259
        OUT 10h            ;ICW1
        MVI A, 0
        OUT 11h            ;ICW2
        MVI A, 01111110B   ;OCW1
        OUT 11h            ; SE HABILITAN LAS INTERRUPCIONES

        ;INICIALIZACIÓN DE LA MEMORIA
        MVI A, 0
        STA MAYOR
        STA MENOR
        STA ACTUAL
        LXI H,2100h        ;HL=2100
        SHLD PUNTERO       ;PUNTERO=2100 (DEJA EL PUNTERO APUNTANDO
                           ;A LA PRIMERA POSICION DE MEMORIA
                           ;PARA RELLENAR)
BUCLE:  JMP BUCLE          ;FIN DEL PROGRAMA PRINCIPAL, BUCLE INFINITO
;*****************************************************************
;INTERRUPCION DEL TEMPORIZADOR.
;*****************************************************************
IRQ7:  PUSH PSW            ;GUARDA EL ACUMULADOR Y LOS FLAGS
       PUSH H              ;GUARDA H Y L
       LXI H, FFF1h        ;HL APUNTAN AL PUERTO B DEL SEGUNDO 8255
       MVI A, 00000001b    ; MÁSCARA PARA CAMBIAR EL BIT0 DEL PUERTO B
       XRI M               ;PONE A 1 EL BIT 0 DEL PUERTO B (ESTABA A 0)
       XRI M               ;VUELVE A PONER A 0 EL BIT 0 DEL PUERTO B
       MVI A, 00100000b    ;código de fin de interrupción
       OUT 10h             ;al 8259
       POP H               ;RECUPERA H Y L
       POP PSW             ;recupera el acumulador y los flags
       RET                 ;fin de la interrupción.
```

```
;*********************************************************************
;INTERRUPCION DEL ADC.
;*********************************************************************
IRQ0:  PUSH PSW          ;GUARDA EL ACUMULADOR Y LOS FLAGS
       PUSH H            ;GUARDA H Y L
       LDA FFF0h         ;LEE EL DATO DESDE EL ADC

       ;GUARDO LA MUESTRA EN MEMORIA
       LHLD PUNTERO      ;HL=PUNTERO
       STAX H            ;COPIA  LA MUESTRA EN MEMORIA
       LHLD ACTUAL       ;HL=ACTUAL
       STAX H            ;COPIA  LA MUESTRA EN “ACTUAL”

       ;INCREMENTO PUNTERO.
       LHLD PUNTERO      ;CARGO EL PUNTERO EN HL
       INX H             ;LO INCREMENTO
       SHLD PUNTERO      ;Y LO ALMACENO DE NUEVO

       ;COMPARO EL PUNTERO CON 25A0
       LDA PUNTERO       ;LEO LSB
       CPI A0h           ;COMPARO CON A0h
       JNZ MAYOR
       LDA PUNTERO+1     ;SI EL LSB ES IGUAL, LEO EL MSB
       CPI 25h           ;Y LO COMPARO CON 25h
       JNZ MAYOR
       LXI H,2100h       ;SI LOS DOS SON IGUALES
       SHLD PUNTERO      ;INICIALIZA EL PUNTERO A 2100h, EL PRINCIPIO DE
                         ;LA MEMORIA DE MUESTRAS
MAYOR: LXI ACTUAL        ;HL APUNTAN A LA DIR DE LA TEMPERATURA ACTUAL
       LDA MAXIMA        ;A=MAX
       CMP M             ; ¿ACTUAL>MAX? (HACE MAX-ACTUAL)
       JP MENOR          ;SI ES POSITIVO (MAX >= ACTUAL) SALTA
       LDA ACTUAL        ;SI NO, ES MAYOR, LEE EL VALOR
       STA MAXIMA        ;Y LO ALMACENA COMO MAXIMA
```

```
MENOR: LDA MINIMA     ;A=MIN
       CMP M          ; ¿ACTUAL<MIN? (HACE MIN-ACTUAL)
       JM DISPLAY     ;SI ES NEGATIVO (MIN <= ACTUAL) SALTA
       LDA ACTUAL     ;SI NO, ES MENOR, LEE EL VALOR
       STA MINIMA     ;Y LO ALMACENA COMO MINIMA

DISPLAY: LDA MINIMA
       STA  FF00h     ;SACA LA TEMPERATURA MINIMA POR EL PUERTO A
       LDA ACTUAL
       STA FF01h      ;LA ACTUAL POR EL PUERTO B
       LDA MAXIMA
       STA FF02h      ;Y LA MÁXIMA POR EL PUERTO C

       OUT 10h        ;al 8259
       POP H          ;RECUPERA H Y L
       POP PSW        ;recupera el acumulador y los flags
       RET            ;fin de la interrupción.
```