

## EJERCICIOS-MICROPROCESADORES

### EJERCICIO-1

Se pretende diseñar un sistema basado en un microprocesador Intel 8086, que controle la temperatura de un invernadero.

Para ello dispone de un ADC de 8 bits, que adquiere los valores de un sensor de temperatura. El rango del sensor es de 0 a 80 °C, generando una tensión proporcional a esos valores de temperatura. Se supone que el rango del conversor es el mismo que el del sensor, por lo que el código 00H se corresponde con 0°C y FFH con 80°C.

El control es de tipo todo-nada con un margen de temperatura alrededor de 30°C, de ±5°C. Es decir, si supera los 35°C hay que desconectar los calefactores, y si cae por debajo de 25°C hay que conectarlos.

Los calefactores (8 en total) poseen una interfaz digital de control (1:ON, 0:OFF). Las muestras de temperatura se tomarán con una frecuencia de 10 Hertzios.

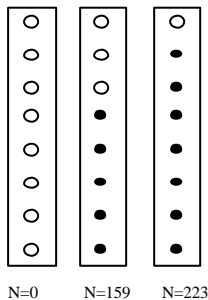
- a.-Diseñar el sistema con los componentes hardware necesarios.
- b.-Diseñar la rutina que controla la temperatura.

### EJERCICIO-2

Se tiene un sistema con un 8086 que dispone de una interface analógica formada por un conversor A/D de 8 bits.

Se quiere construir un indicador luminoso de nivel de señal. Estará formado por un conjunto de 8 leds, que se irán iluminando de forma proporcional al nivel de la señal analógica de entrada, (por ejemplo una señal de audio). Según aumenta el nivel de la señal de entrada, se encenderán más leds, desde nivel 0 voltios (ningún led iluminado), hasta 2 voltios (todos los leds iluminados).

Ejemplo para N = 0, 159, 223



Si N es el código que devuelve el conversor. La relación entre la tensión de entrada y el código N es la siguiente:

$$N = \text{ParteEntera} \left[ \frac{V_{in}}{V_{max}} \cdot 255 \right]$$

$V_{in}$ : tensión de entrada (0 a 2 voltios)

$V_{max}$ : tensión máxima admitida por el conversor (2 voltios)

Para calcular el valor que ha de representarse en los leds:

El valor N+1 se divide por 32 (256 códigos / 8 leds), que es el incremento que debe sufrir el código del conversor para iluminar un led más. Luego se eleva dos, al resultado anteriormente obtenido. Por último se decrementa en uno:

Ej:  $V_{in}=1$  v, luego  $N=127$

$$\begin{aligned}
 127 + 1 &= 128 \\
 128 / 32 &= 4 \text{ (hay que encender los 4 leds más bajos)} \\
 2^4 - 1 &= 15 \rightarrow \text{RESULTADO}
 \end{aligned}$$

El convertor analógico-digital tiene el registro de datos mapeado en la dirección 300h de E/S. Se utiliza una lógica de decodificación para activar la señal de selección del convertor. Las direcciones de E/S 301H-31FH se encuentran disponibles.

Se pide:

a.-Hacer un esquema del interface de E/S incluyendo los dispositivos que se estime necesario para controlar los leds. Basta con un diagrama de bloques, no siendo necesario detallar la lógica de decodificación.

b.-Escribir el programa en Ensamblador del 8086 que, en un bucle infinito, esté leyendo el valor de las muestras en el convertor y controlando los leds.

### **EJERCICIO-3**

Se tiene un sistema microprocesador basado en un 8086 para el control de temperatura y humedad de un invernadero. El sistema tiene una interfaz analógica-digital compuesta por 2 convertidores A/D de 8 bits.

Aunque este control no exige una temporización estricta (temperatura y humedad varían de forma muy lenta), se decide implementar las rutinas con interrupciones. El muestreo se hace a una frecuencia de 400 Hz (temperatura) y 200 Hz (humedad), por lo que el sistema dispone de un temporizador que genera 2 señales que activan dos de las entradas de un controlador de interrupciones. Este generará interrupciones enmascarables, cuyos tipos son el 8 y el 9. Es necesario enviarle al PIC el comando EOI al terminar la rutina de interrupción (20h a la dirección 20h de E/S).

Los registros de datos de los convertidores están mapeados en la direcciones de E/S: 300h (temp.) y 302h (humedad).

Control de temperatura.-Cuando la temperatura alcanza un valor igual o superior a  $T_{max}$ , se debe encender un refrigerador. Cuando la temperatura sea inferior a  $T_{min}$ , un calefactor.  $T_{max}$  y  $T_{min}$  son temperaturas límite que marcan el margen de temperaturas permitido. Suponer que  $T_{max}$  se corresponde con el código T1 del convertor, y  $T_{min}$  con T2. Tanto el calefactor como el refrigerador, disponen de una interfaz digital compuesta por una señal de control: encendido/apagado (1:ON, 0:OFF).

Para el control de humedad se utiliza un algoritmo parecido, con otros dos niveles de humedad, que se corresponden a H1 (valor max) y H2 (valor min). En este caso, una electroválvula controla el paso de agua por una tubería. El control se hace según un bit (1:abierta, 0:cerrada).

Se pide:

a.-Esquema general del sistema.

b.-Diseñar la interface digital. Funciones de selección, lógica de selección y esquema. Las direcciones libres del micro son 300-31f (menos las de los convertidores (A/D). Utilizar una pal para decodificar.

c.-Diseñar el programa.

#### **EJERCICIO-4**

Se tiene un sistema microprocesador basado en un 8086 para el tratamiento de señales de audio. El sistema tiene una interfaz analógica/digital compuesta por un convertidor A/D y D/A de 8 bits. El muestreo se hace a una frecuencia de 8 KHz, por lo que el sistema dispone de un temporizador que genera una señal que activa una de las entradas de un controlador de interrupciones. Este generará una interrupción no enmascarable al micro, cuyo tipo es el 8. Es necesario enviarle el comando EOI al terminar la rutina de interrupción (20h a la dirección 20h de E/S).

El sistema implementa un algoritmo de generación de ecos. El valor de salida hacia el DAC debe calcularse como:

$$y[n] = x[n] + 0.25x[n-4000] \quad \text{NOTA: } 0.25=25/100$$

Es decir, la salida es la muestra natural (índice n) más la muestra atenuada en un instante anterior (4000/8000= 0.5 segundos antes).

Por tanto, será necesario tener previsto un buffer para almacenar 4001 muestras. Cada vez que se obtiene una muestra hay que calcular la salida y desplazar todas las muestras.

Los registros de datos de los conversores están mapeados en las direcciones de E/S: 300h (A/D) y 320h (D/A).

Se pide diseñar el programa en Ensamblador para el 8086 que sea capaz de realizar el algoritmo.

#### **EJERCICIO-5**

Se tiene un sistema microprocesador basado en un 8086 para el control de un aparcamiento.

Cada vez que llega o sale un coche, se activa un sensor que indica la presencia de un coche en la barrera. Si hay plazas suficientes, se levanta la barrera de entrada. Cada vez que llega un coche a la barrera de salida, se levanta dicha barrera y se actualiza el contador de coches en el aparcamiento.

Se dispone de un indicador luminoso (de tipo 7 segmentos), que muestra el número de plazas libres del aparcamiento (0 a 999) en cada momento.

El sistema dispone de un controlador PIC, al que se le conectan los sensores de presencia de las barreras y un temporizador para la multiplexación de displays. Se corresponden con las interrupciones de tipos 6, 7 y 8. De esta manera, cada vez que llega un coche a una barrera (entrada o salida) se activa una interrupción. Es necesario enviar el comando EOI al PIC para finalizar la interrupción correctamente (20H a la dirección 20H de E/S).

Las barreras poseen una interfaz digital para su control (1: abrir, 0: cerrar). Para cerrar con seguridad, existe un sensor por cada barrera, que indica que el coche ya ha pasado la barrera y ésta, por tanto, se puede cerrar. Dicho sensor proporciona una señal digital (1: presencia, 0: el coche ha rebasado la barrera) que no está conectada al PIC.

El esquema general se encuentra en la parte posterior del enunciado.

Se pide:

- a.-Hacer las conexiones oportunas en el esquema general.
- b.-Diseñar la interfaz digital. Funciones de selección, y dispositivos utilizados. Las direcciones libres del microprocesador son 300-31f. Para el sistema de decodificación indicar sólo variables de entrada y señales de salida.
- c.-Diseñar las 2 rutinas de atención a las interrupciones de las barreras.

si plazas >0 :    abrir barrera  
                          esperar al sensor de seguridad.  
                          cerrar barrera  
                          actualizar plazas

abrir barrera  
esperar sensor de seguridad  
cerrar barrera  
actualizar plazas

*Barrera de entrada*

*Barrera de salida*

d.-Diseñar la rutina que controla los displays 7 segmentos. Esta rutina se encargará, de forma periódica, de leer una variable global (plazas) y representar en los displays dicha información. Se dispone de una rutina auxiliar ('bcd'), ya implementada, que devuelve en un buffer de memoria 3 bytes consecutivos con los 3 dígitos BCD. A esta rutina se le pasan como parámetros la dirección del buffer (offset) y el número a convertir.