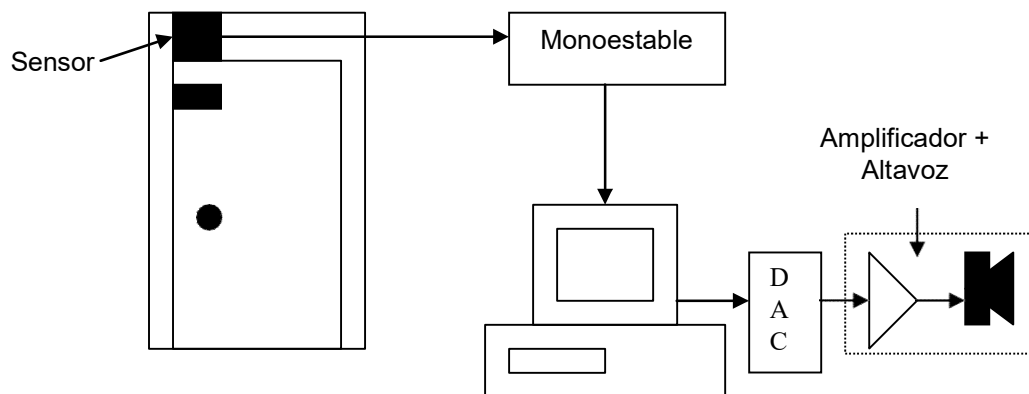


SISTEMAS BASADOS EN MICROPROCESADORES (SBM)
Grado de Ingeniería Informática
Doble Grado Informática-Matemáticas

Escuela Politécnica Superior – UAM
FINAL MAYO - Curso 17-18
ENUNCIADO

Diseñar e implementar el código de un sencillo sistema de alarma basada en un ordenador personal compatible PC. El sistema consta de un sensor magnético (cápsula REED con dos láminas metálicas que se tocan en reposo) que se comporta como un microinterruptor accionado por el campo magnético de un pequeño imán. Este sensor se puede colocar en puertas o ventanas, situando el imán en la propia puerta (u hoja de la ventana) y el microinterruptor (cápsula) en el marco de la misma. Cuando la puerta está cerrada, el imán se encarga de mantener abierto el microinterruptor (separa las dos láminas), que se cerrará al abrirla, al alejar el imán del mismo. El cierre del microinterruptor activa un sencillo circuito monoestable, que generará un **impulso negativo** de la duración adecuada.



Un programa será el encargado de detectar la activación del sensor y de reproducir un mensaje pregrabado a través de un sencillo sistema de audio, compuesto de un conversor digital-analógico (DAC) de 8 bits, un amplificador y un altavoz. El mensaje de voz ha sido grabado con una frecuencia de muestreo de 8 KHz, y 8 bits por muestra. La velocidad de reproducción de las muestras debe ser lo más próxima posible a esa frecuencia de muestreo para no alterar las características del mensaje. Para controlar la frecuencia de reproducción del mensaje, utilizaremos el RTC, convenientemente configurado para generar 8000 interrupciones por segundo, de forma que, con cada interrupción, se enviará una muestra (8 bits) al conversor DAC de 8 bits. El conversor DAC permitirá reconstruir el mensaje (digital) de 5 segundos grabado previamente, generando una señal analógica que será amplificada por el sistema de audio. El conversor DAC (8 bits de entrada) estará conectado a los 8 bits de datos del puerto LPT1 (D7-D0) y la salida del monoestable estará conectada al pin ACK# de dicho puerto para que la activación del sensor genere una interrupción en el LPT1, que indicará a través de una variable ALARMA (su valor pasa a 1) que se ha activado el sensor. Cuando la variable ALARMA sea "1", empezará la reproducción del mensaje utilizando otra variable global CONTADOR. El mensaje grabado digitalmente como un buffer de valores (muestras) de 8 bits sin signo tiene una duración de 5 segundos, luego se reproducirán 40.000 muestras. Para resetear el flag de alarma, será necesario resetear el hardware para que vuelva a armarse la alarma, borrándose

el flag. Mientras el flag de alarma esté activo (variable ALARMA a "1"), el mensaje se reproducirá repetidamente hasta que se reinicie el sistema.

A continuación se incluye el detalle del segmento de datos (variables) de las que hemos hablado en el enunciado y que deberán utilizarse en la implementación del código en las preguntas donde sea necesario.

;Zona de variables

;Mensaje de alarma

DATOS

SEGMENT

MENSAJE DB 10,15,127,23,45,86, ..., 114 ;40.000 muestras

CONTADOR DW 0 ;Cuenta las muestras reproducidas

ALARMA DB 0 ;Flag que indica activación o no del sensor

DATOS ENDS

El código de la aplicación estará formado por el programa principal **PPAL (P1)** y el código de las rutinas **DETALARM (P2)** y **ENVIOMST (P3)**.

SISTEMAS BASADOS EN MICROPROCESADORES (SBM)
Grado en Ingeniería Informática y Doble Grado Informática-Matemáticas
Escuela Politécnica Superior – UAM
FINAL MAYO - Curso 17-18
SOLUCIÓN PREGUNTAS

NOMBRE: _____ **DNI :** _____

APELLIDOS: _____

P1. Teniendo en cuenta el enunciado del ejercicio y el código incluido en el mismo, escriba el código del programa principal. Añada los comentarios que considere oportunos para mayor claridad del código implementado. (4 p)

```
PPAL      PROC FAR
          ASSUME DS: DATOS

          MOV  AX, DATOS
          MOV  DS, AX
          CLI                                     ;Inhibir las interrupciones CPU
          XOR  AX, AX
          MOV  ES, AX
          ;Inicializar vector 70H
          MOV  ES:[70H*4], OFFSET ENVIOMST
          MOV  ES:[70H*4 + 2], SEG ENVIOMST
          ;Inicializar vector 0FH
          MOV  ES:[0FH*4], OFFSET DETALARM
          MOV  ES:[0FH*4+2], SEG  DETALARM
          STI                                     ;Habilitar las interrupciones CPU
          MOV  DX, ES:[0408h]
          ADD  DX, 2
          IN   AL, DX                             ;Activar IRQEN
          OR   AL, 10H
          OUT  DX, AL
          MOV  AL, 0AH
          OUT  70H, AL                             ;Programar RTC (8Khz)
          MOV  AL, 23H
          OUT  71H, AL
          MOV  AL, 0BH
          OUT  70H, AL                             ;Programar IRQ de RTC
          IN   AL, 71H
          OR   AL, 40H
          MOV  AH, AL
          MOV  AL, 0BH
          OUT  70H, AL
          MOV  AL, AH
          OUT  71H, AL
          MOV  ALARMA, 0                          ;Se pone a "0" el flag de alarma
          IN   AL, 21H                             ;Habilita interr. PIC-0 (IR7)
          AND  AL, 7FH
          OUT  21H, AL
          IN   AL, 0A1H                             ;Habilita interr. PIC-1 (IR0)
          AND  AL, 0FEH
          OUT  0A1H, AL

          BUCLE:
          JMP  BUCLE                               ;Bucle del programa principal
PPAL      ENDP
```

P2. Teniendo en cuenta el enunciado del ejercicio, escriba el código de la rutina DETALARM. Añada los comentarios que considere oportunos para mayor claridad del código implementado. (3 p)

```

;Rutina de servicio de la interrupción 0Fh
DETALARM    PROC FAR
             STI                ;Habilitar las interrupciones
             PUSH AX
             MOV  ALARMA, 1     ;Se pone a "1" un flag de alarma
             MOV  AL, 20H
             OUT  20H, AL       ;Enviar un EOI al PIC-0
             POP  AX
             IRET
DETALARM    ENDP

```

P3. Teniendo en cuenta el enunciado del ejercicio, escriba el código de la rutina ENVIOMST. Añada los comentarios que considere oportunos para mayor claridad del código implementado. (3 p)

```

;Rutina de servicio de la interrupción 70h
ENVIOMST    PROC FAR
             STI                ;Habilitar las interrupciones
             PUSH AX SI DX

             MOV  AL, 0CH
             OUT  70H, AL        ;Leer el registro C
             IN   AL, 71H
             CMP  ALARMA, 1
             JNE  SALIR
             MOV  SI, CONTADOR
             MOV  AL, MENSAJE[SI]
             MOV  DX, ES:[0408h] ;Dirección base de LPT1
             OUT  DX, AL         ;Envía muestra a LPT1
             INC  CONTADOR       ;Incrementa el contador
             CMP  CONTADOR, 40000
             JNE  SALIR
             MOV  CONTADOR, 0    ;Inicializa contador

SALIR:
             ;Enviar un EOI (20H) a cada PIC
             MOV  AL, 20H
             OUT  20H, AL
             OUT  0A0H, AL

             POP  DX SI AX
             IRET
ENVIOMST    ENDP

```