

2.8 MÓDULO DE INTERRUPCIÓN POR TECLADO – KBI AÑADIENDO INTERRUPTORES A SU MICROCONTROLADOR

Preparado por: Rangel Alvarado
Estudiante Graduando de Lic. en Ing. Electromecánica
Universidad Tecnológica de Panamá
Panamá, Panamá
“e-mail”: issaiass@cwpanama.net
“web site”: <http://www.geocities.com/issaiass/>

ÍNDICE

| | | |
|-------|--|-----|
| 2.8.1 | Introducción | 274 |
| 2.8.2 | El rebote en Dispositivos Mecánicos | 275 |
| 2.8.3 | Registros del Módulo de Interrupción por Teclado | 275 |
| 2.8.4 | Diagrama de Flujo | 277 |
| 2.8.5 | Código | 278 |
| 2.8.6 | Simulación | 283 |
| 2.8.7 | Conclusión | 284 |
| 2.8.8 | Referencias | 284 |
| 2.8.9 | Problemas Propuestos | 284 |

2.8.1 Introducción

Todo usuario necesita insertar condiciones en un sistema, como por ejemplo, inicializar el tiempo de un cronómetro digital, controlar el sentido de dirección de un sistema, iniciar o detener un motor y el teclado es el método de control más elemental.

Algunos microcontroladores de la familia HC08 pueden reconocer cuando una tecla fue presionada y generar interrupciones¹. El módulo de teclado proporciona:

- Siete (7) posibles interrupciones independientes
- Configurar cada puerto con una resistencia a cinco (5) voltios (“pull-ups”).
- Permitir que solo se reconozca una o varias teclas presionadas al mismo tiempo.
- Salida de modos de bajo consumo como espera (wait) o parada (stop).

Con el solo propósito de demostrar tal captura de teclado se describe en este documento a:

- Configurar una tecla de interrupción del módulo de teclado: Se configura la tecla PTA5 y PTA0 para capturar solo la tecla presionada.
- Generar una interrupción: La tecla configurada generará una interrupción y conmutará el LED de la tarjeta conectado al puerto PTD7.
- Simular: ver por medio de breakpoints donde se encuentran ubicados los registros correspondientes en el mapa de memoria al simular.

¹ p.e. para el microcontrolador JL3 los puertos PTA[0:6] pueden utilizarse para conectar un teclado.

2.8.2 El Rebote en Dispositivos Mecánicos

El rebote es una característica inherente de dispositivos mecánicos, obviamente no deseada en sistemas digitales. Dichos rebotes no son más que estados inciertos entre encendido y apagado hasta llegar a la estabilidad de la señal. Una de las maneras más comunes de eliminar el rebote es retardar unos tantos milisegundos luego de una lectura y comprobar nuevamente el estado de la tecla contra el estado anterior.

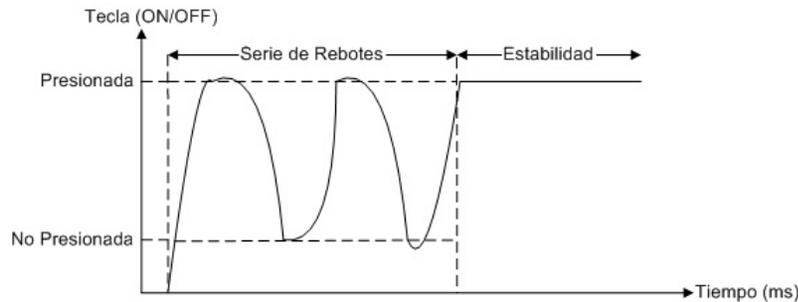
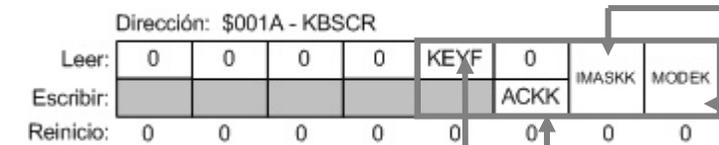


Figura 160. Rebotes Mecánicos. Cualquier pieza mecánica que se presione o no francamente, puede producir rebotes que son transiciones entre picos máximos y mínimos hasta llegar a la estabilidad de la señal.

2.8.3 Registros del Módulo de Interrupciones por Teclado

2.8.3.1 Registro de Estado y Control del Teclado

R = Reservado
 = No implementado



1 = Alguna tecla fue presionada, interrupción pendiente
 0 = No hay interrupciones pendientes
 Si se escribe un 1 borra las interrupciones pendientes, pero siempre que se lea, leerá un 0

1 = Si existe una interrupción a procesar, no se lleva a cabo (enmascara).
 0 = Si existe la interrupción, el sistema la procesa (no enmascara).

1 = Generará interrupciones si cualquier tecla es presionada o todavía se mantiene presionada.
 0 = No generará interrupciones si existe todavía una tecla presionada.

Nota: **MODEK = 1, genera una señal de ACKK = 1 en la interrupción.**
Se recomienda siempre al final de la interrupción tener ACKK = 1.

Figura 161. Registro de Estado y Control del Módulo de Interrupción por Teclado. Controla el estado de detección dependiendo si la tecla es presionada o si hay ambas presionadas (MODEK); borra (ACKK) las interrupciones que están pendientes (KEYF) o pueden desactivarse vía "software" (IMASKK).

2.8.3.2 Registro de Habilitación de Teclas Interruptoras del Teclado

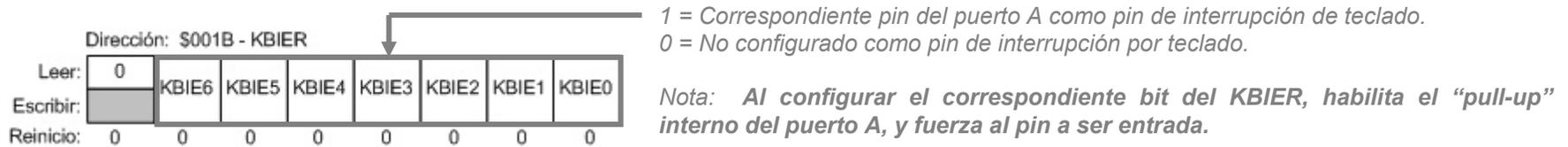


Figura 162. Registro Habilitador de Teclas Interruptoras del Teclado. El correspondiente bit, en uno (1), destina al pin del puerto como interruptor de teclado.

2.8.3.3 Interrupción del Teclado

Si las interrupciones de teclado no han sido enmascaradas (IMASKK), se generará una interrupción cada vez que se presione una tecla dependiendo de la selección del bit MODEK.

Tabla 63. Vectór de Interrupción del Teclado

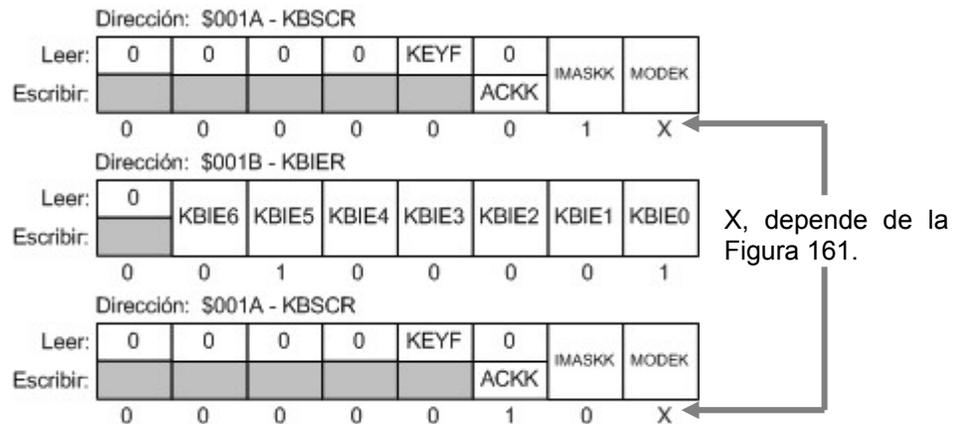
| Bandera | Máscara | Dirección | Vector (Dirección) |
|---------|---------|-----------|--------------------|
| KEYF | IMASKK | FFF0 | Teclado (Alto) |
| | | FFF1 | Teclado (Bajo) |

Figura 163 (inferior-derecha). Pasos de Inicialización del Teclado. Para evitar falsas interrupciones, se recomienda seguir una de las rutinas de inicialización del teclado.

R = Reservado
 [Shaded Box] = No implementado

2.8.3.4 Inicialización de Teclado

Como la señal tarda en estabilizarse al voltaje lógico, se tiene que inicializar el módulo de teclado para preveer una falsa interrupción. Configure las teclas cero (0) y cinco (5) para inicializar el módulo de teclado:



2.8.4 Diagrama de Flujo

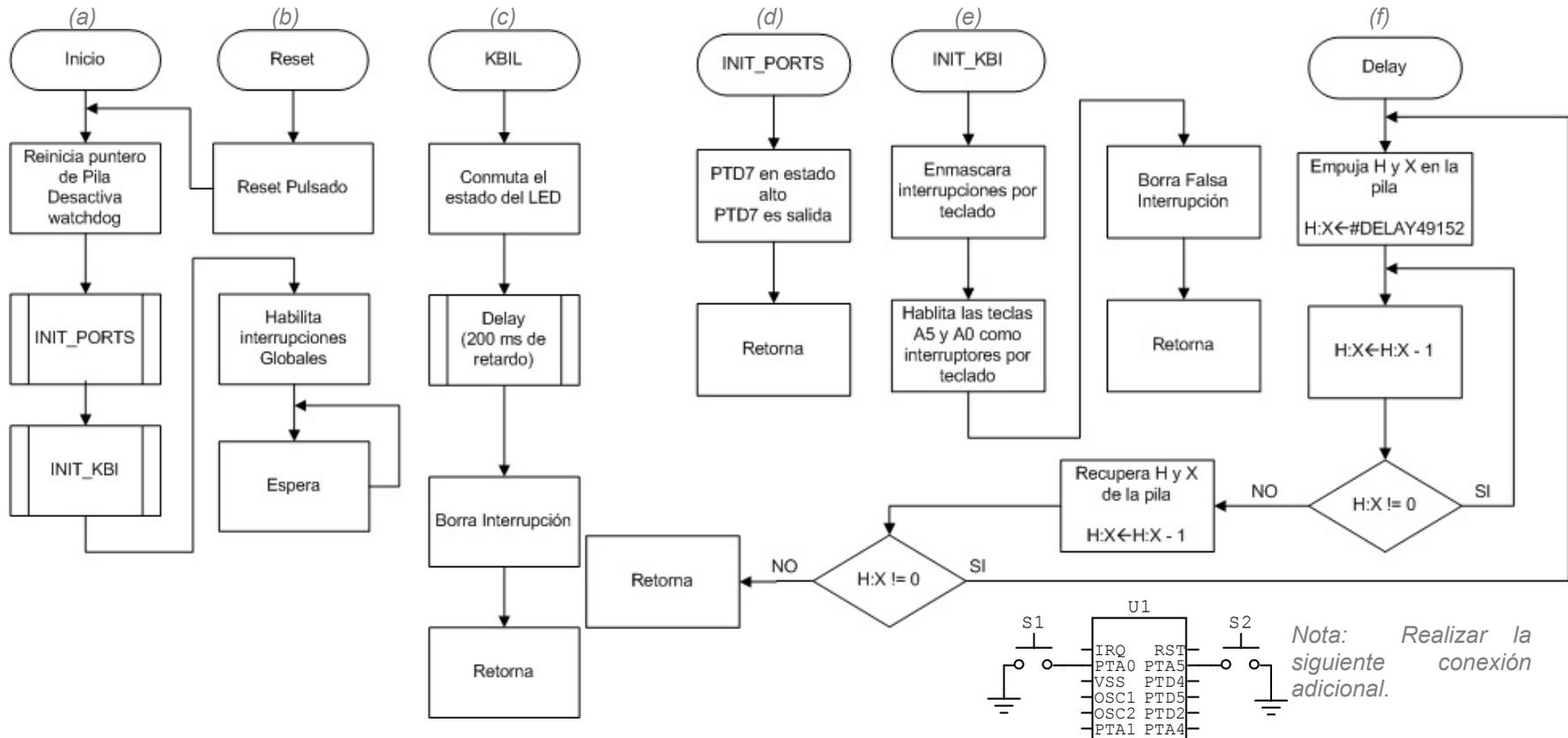


Figura 164. NT0108 – KBI. (a) Programa Principal. Inicializa el módulo de teclado, teclas A5 y A0, puerto D7 como salida. (b) Reinicio del Sistema. Al presionar “Reset”, el sistema es reiniciado sin importar su estado actual. (c) Interrupción de teclado. Conmuta el estado del LED PTD7 y retarda doscientos milisegundos (200 ms) de eliminación de rebote. (d) Inicialización de Puertos. Puerto D7 es una salida en estado alto. (e) Inicialización de Teclado. Inicializa el teclado para eliminar falsas interrupciones del Puerto A5 y A0. (f) Retardo. Subrutina de retardo programable para eliminación de rebotes.

2.8.5 Código

```

=====
;
; ARCHIVO      : NT0108 - Teclado - 09 07 04.asm
; PROPÓSITO   : Generar una interrupción de teclado cada vez que se presio-
;               ne solamente la tecla PTA0 ó PTA5 y conmuta el LED de la
;               tarjeta conectado al puerto PTD7.
;
; NOTA        :
;               1 - Observar la rutina de inicialización de Teclado.
;               2 - Añadir interrupciones de Teclado.
;
; REFERENCIA:
;               Advanced Information of MCU68HC908JK1, JK3, JL3...
;               http://www.freescale.com/files/microcontrollers/
;               doc/data_sheet/MC68HC08JL3.pdf
;               Pág. 155 @ 161 - Módulo de interrupción por teclado
;
; LENGUAJE    : IN-LINE ASSEMBLER
;
;-----
; HISTORIAL
; DD MM AA
; 26 05 03 Creado.
; 10 09 04 Modificado.
;
;-----
; Pasos para iniciar una KBI:
;
;               1 - Enmascarar interrupciones por teclado (IMASKK = 1)
;               2 - Habilitar la tecla correspondiente como interrupción de
;               teclado (KBIEX = 1)
;               3 - Reconocer la falsa interrupción (ACKK = 1)
;               4 - Borrar la máscara de Teclado (IMASKK = 0)
;
; Otra manera:
;               1 - Configurar los pines como salida (DDRAX = 1)
;               2 - Enviar el correspondiente pin a 1 lógico (PTAX = 1)
;               3 - Habilitar la tecla correspondiente como interrupción de
;               teclado (KBIEX = 1)
;
;-----
$SET   ICS08                ; ICS08 = 1, Vamos a simular en la pastilla
;                               ; la velocidad de simulación es menor en la
;                               ; PC.
;
;$SETNOT ICS08              ; ICS08 = 0, Vamos a programar la pastilla
;                               ; la aplicación debe correr en tiempo real

```

```

;=====
;
;                               Definiciones del Usuario
;=====
COPD      equ 0T                ; Bit 0 del registro CONFIG1
PTD7      equ 7T                ; Puerto D, Bit 7
DDRD7     equ 7T                ; Registro de Direccionamiento D, Bit 7
ACKK      equ %00000100        ; KBSCR, Bit de Reconocimiento de
;                               Interrupción, Bit 2 ON
IMASKK    equ %00000010        ; KBSCR, Bit de Máscara del Registro
;                               KBSCR, Bit 1 ON
MODEK     equ %00000001        ; KBSCR, Bit de Selección de Detección, Bit 0
;                               ON
KBIE5     equ %00100000        ; KBIER, Bit de Interrupción, Bit 5 ON
KBIE0     equ %00000001        ; KBIER, Bit de Interrupción, Bit 0 ON
DELAY49152 equ $0096           ; Constante de Retardo
MS200     equ $00C8            ; Constante de 200 ms de retardo

;=====
;
;                               Mapa de Memoria del Microcontrolador
;=====
;=====
;                               Registro de E/S
;=====
PORTD     equ $0003            ; Dirección, Puerto D
DDRD      equ $0007            ; Registro de Direccionamiento, Puerto D

;=====
;
;                               Módulo de Interrupción por Teclado
;=====
KBSCR     equ $001A            ; Registro de Estado y Control del KBI
KBIER     equ $001B            ; Registro de Control de Interrupciones de
;                               Teclado

;=====
;
;                               Registro de Configuraciones
;=====
CONFIG1   equ $001F            ; Vectores de configuración

;=====
;
;                               Memoria FLASH
;=====
FLASH_START equ $EC00          ; Puntero - Mem.FLASH

;=====
;
;                               Vectores de Usuario
;=====
KBINH     equ $FFE0            ; Interrupción de Teclado
RESET_VEC equ $FFFE            ; Puntero del RESET

```

```

;=====
; OBJETIVO   : Inicio de Codif. del Ensam-
;             : blador en Memoria FLASH.
;=====
                org FLASH_START                ; Inicio Mem. FLASH

;=====
; OBJETIVO   : Genera una interrupción de
;             : tecla cada vez que se pre-
;             : sione uno de los dos botones
;             : configurados, si existe to-
;             : davía una tecla en bajo o
;             : ambas, no llama la inte-
;             : rrupción
;=====
START
    rsp                ; Inic.Stack = $00ff
    bset COPD,CONFIG1 ; Desactiva watchdog
    clra               ; Borra A
    clrx               ; Borra X
    jsr INIT_PORTS    ; Inicializa Puertos
    jsr INIT_KBI      ; Inicializa Módulo de Teclado
    cli               ; Habilita Interrupciones
ESPERA
    wait              ; Espera la interrupción.
    bra ESPERA       ; Salta al modo de bajo consumo

;=====
; INIT_PORTS : Inicializa variables y regis-
;             : tros.
; OBJETIVO   : Inicializa los registros de
;             : direccionamiento.
;             : PORTD7 = HIGH
;             : PORTD7 = OUTPUT
; ENTRADA    : Ninguna
; SALIDA     : Ninguna
; REGISTROS
; AFECTADOS : DDRD, PORTD
;=====
INIT_PORTS
    bset PTD7,PORTD   ; Puerto D 7 en alto
    bset DDRD7,DDRD   ; Puerto D 7 es salida
    rts               ; retorna

```

```

=====
;
; INIT_KBI      : Inicializa registro de inte-
;                rrupción de teclado.
;
; OBJETIVO     : Teclas PTA0 y PTA5 como inte-
;                rruptores de teclado.
;
; ENTRADA      : Ninguna
; SALIDA       : Ninguna
; REGISTROS     :
; AFECTADOS    : KBSCR, KBIER
;
=====
INIT_KBI
    mov #IMASKK,KBSCR      ; Enmascaro las interrupciones por
                           ; teclado
    mov #{KBIE5|KBIE0},KBIER ; Escribo en KBIER y habilito las teclas
                           ; Se generó una falsa interrupción.
    mov #ACKK,KBSCR       ; Borro la falsa interrupción
    rts                   ; Retorno de la subrutina.

=====
;
; DELAY        : Genera un retardo de tiempo
; OBJETIVO     : Retardo de tiempo, base 1ms
; ENTRADA      : H:X = Retardo en ms
; SALIDA       : H:X = 0
; REGISTROS     :
; AFECTADOS    : H:X
; USO          :
;
;                MIN = H:X = 1T
;                MÁX = H:X = 65535T
;                ldhx #500
;                jsr Delay ; retarda 0.5 seg
;
=====
Delay  pshx                ; [2] Salva X en la pila
      pshh                ; [2] Salva H en la pila
      ldhx #DELAY49152    ; [3] Carga constante de bucle fino
Delay0 aix #-1            ; [2] Decrementa H:X en 1
      cphx #0             ; [3] Llegó a cero (0)
      bne Delay0         ; [3] Si no es igual, salta a Delay0
      pulh               ; [2] Si es igual, recupera H de la pila
      pulx               ; [2] Recupera X de la pila
      aix #-1            ; [2] Decrementa H:X en 1
      cphx #0           ; [3] Llegó a cero (0)
      bne Delay         ; [3] Si no es igual, salta a Delay
      rts                ; [4] retorna

```

```

;=====
;KBIL      : Interrupción de teclado
;OBJETIVO  : Reconoce la interrupción al
;           presionarse una tecla, si se
;           mantiene presionada, no gene-
;           rará una siguiente interrup-
;           ción hasta que se suelte la
;           tecla.
;ENTRADA   : Ninguna
;SALIDA    : Ninguna
;REGISTROS
;AFECTADOS : KBSCR, KBIER, PORTD, A, H:X
;=====
KBIL
    lda PORTD          ; Lee el puerto
    coma              ; cambia 1's por 0's
    sta PORTD         ; PORTD = A
    $IFNOT ICS08      ; Si no se va a simular, el eliminador de
                    ; rebotes se encuentra activo
        ldhx #MS200   ; A retardar 200 ms
        jsr Delay     ; Retarda 200 ms
    $ENDIF            ; Fin de la compilación condicional
    mov #ACKK,KBSCR   ; Borra interrupciones subsiguientes (ACKK =
                    ; 1).
    rti               ; Retorno de la interrupción.

;=====
;OBJETIVO  : Inicializa el Vector de Reset
;           Arranque del programa en la
;           memoria Flash.
;=====
;===== Vectores de Teclado, Tecla presionada =====
    org KBIH          ; Vector de Teclado
    dw KBIL           ; Tecla apretada
;===== Vector de Reinicio de Sistema =====
    org RESET_VEC     ; Puntero Vec - RESET
    dw START          ; al darse reset salta a Start

```

Listado 17. NT0108 - KBI. El programa responde a inicializar el módulo de teclado y cada vez que se presiona, se generará una interrupción la cual conmutará el LED de la tarjeta conectado al puerto PTD7.

2.8.6 Simulación



Figura 165. Simulación del Programa Principal. (a) "Breakpoint". Sitúe un "breakpoint" para visualizar la configuración de pines de teclado. (b) Rutina de Inicialización de Teclado. Note que pasos hay que seguir para inicializar un módulo de teclado. (c) Interrupción. Genere una interrupción presionando una tecla y observe que pasa si la tecla queda presionada.

- (a) Inicie WinIDE.
- (b) Cargue el archivo NT0108 – KBI – 09 07 04.asm.
- (c) Compile.
- (d) Entre al simulador.
- (e) Añada "breakpoints" según la figura 165(a).
- (f) Corra su programa y en la sección de llamado a la rutina de inicialización del teclado (INIT_KBI) observe la ejecución de cada instrucción.
- (g) Presione una tecla y espere a la llegada del "breakpoint". Observe el estado del LED; ejecute dos pasos y observe nuevamente que sucede
- (h) Genere otra interrupción, pero esta vez mantenga la tecla presionada. La interrupción no se repite hasta que se suelte una o todas las teclas.
- (j) Si desea "quemar" su pastilla, revisar la NT0009, Sección 1.9.5.

Nota: Recuerde mover su compilación condicional si desea que su programa corra en tiempo real:

```

; $SET ICS08
$SETNOT ICS08
    
```

2.8.7 Conclusión

El módulo de interrupción por teclado es un aditivo de los microcontroladores de algunos miembros de la Familia HC08 que permite responder a la acción de teclas presionadas para que el usuario tenga la comodidad de adaptar un teclado al sistema.

En esta ocasión, se generó un simple teclado de dos interruptores el cual solo respondía a una sola tecla presionada, esta característica es importante si vamos a pulsar un código de acceso (no queremos doble situaciones).

Para la mayoría de sistemas embebidos, el conectar un teclado no es tan común. Lo más frecuente es ver un par de teclas en dispositivos embebidos y no un teclado de 101 teclas o más, pues con una sola tecla se puede tener múltiples funciones, prueba está en medidores de temperatura o equipos médicos digitales medidores de presión arterial.

2.8.8 Referencias

2.8.8.1 Información Avanzada sobre el Microcontrolador

(a) http://www.freescale.com/files/microcontrollers/doc/data_sheet/MC68HC08JL3.pdf

Págs. 155 a 161 – Módulo de Interrupción Por Teclado (KBI).

2.8.8.2 Manual de Referencia del CPU

(a) http://www.freescale.com/files/microcontrollers/doc/ref_manual/CPU08RM.pdf

2.8.8.3 Funcionamiento del Teclado de una PC

(a) <http://usuarios.lycos.es/labdeelectronica/teclado.htm>

2.8.8.4 Página “web” sobre esta Nota Técnica

(a) <http://www.geocities.com/issaiass/>

2.8.9 Problemas Propuestos

2.8.9.1 Genere un contador de pulsos por medio del módulo de teclado.

2.8.9.2 Inicialice el módulo de teclado y realice un menú que permita tener diferentes secuencias de LEDs dependiendo de la tecla presionada.

2.8.9.3 Utilice el módulo de teclado para que si son presionadas dos teclas al mismo tiempo, apague un LED.