


# PRÁCTICA 3

## Interrupciones por Hardware

 **Objetivos:** Comprender la utilidad de las interrupciones por hardware y el funcionamiento del Controlador de Interrupciones Programable (PIC). Utilizar el TIMER, F10, y el dispositivo de handshaking (HAND-SHAKE) mediante interrupciones para el intercambio de información entre el microprocesador y el mundo exterior.

### Parte 1: PIC y Vector de Interrupciones

- 1) **Registros del PIC** ★ Dados los siguientes registros del PIC indique su función. Además, para el valor de ejemplo, indique qué significa en términos del funcionamiento del programa

Dirección	Registro	Función	Valor de Ejemplo
20h	EOI		No aplica
21h	IMR		1111 1101
22h	IRR		0000 0011
23h	ISR		0000 0100
24h	INT0		15
25h	INT1		10
26h	INT2		25

- 2) **Configuración del PIC** ★ Se desean configurar algunas interrupciones mediante el PIC. Complete la tabla indicando en cada caso los registros o direcciones que faltan para que funcionen las interrupciones, o el dispositivo que se está usando.

Dispositivos	IMR	INT0	INT1	INT2	Dirección Vector
F10	1111 1110	5	No importa	No importa	20
F10					32
	1111 1101				40
HANDSHAKE				10	
	1111 1111				
F10 y Timer		10	20		
	1111 1001				16 y 32
	1111 1000	5	10	15	

**Nota:** En el caso de haber más de un dispositivo en uso, indicarlos en el mismo orden en las columnas "Dispositivos" y "Dirección Vector". En el caso en que una columna no deba configurarse, indicarla como "No importa".

## Parte 2: Interrupciones con la tecla F10, Timer y Handshake

- 3) **Contador de pulsaciones** ★ El siguiente programa cuenta el número de veces que se presiona la tecla F10 y acumula este valor en el registro DX. El programa nunca termina, ya que ejecuta un loop infinito. Completar las instrucciones faltantes para que el programa funcione correctamente.

EOI EQU 20H	<b>ORG 2000H</b>	<b>ORG 3000H</b>
IMR EQU 21H	CLI	<b>RUT_F10:</b> PUSH AX
_____	MOV AL, 0FEH	_____
N_F10 EQU 15	_____	MOV AL, EOI
_____	OUT INT0, AL	_____
IP_F10 DW <b>RUT_F10</b>	MOV DX, 0	POP AX
	STI	IRET
	<b>LAZO: JMP LAZO</b>	END

Luego de completarlo, verifique su funcionamiento en el simulador, y explicar detalladamente:

- ¿Qué hacen las instrucciones CLI y STI? ¿Qué podría suceder si no están las mismas y la persona que usa el programa presiona F10 rápidamente durante la configuración del PIC?
- ¿Por qué se usa el valor 0FEH en el programa?
- ¿Qué instrucciones le indican al PIC que la interrupción ha terminado?
- ¿En qué dirección se encuentra la subrutina que atiende las interrupciones del F10?

4) **Tecla F10 con pantalla y terminación** ★★

A) Escribir un programa que muestre en pantalla “Vamos las interrupciones!” cada vez que se presiona la tecla F10. El programa nunca termina.

B) Idem A pero ahora el programa termina luego de mostrar 5 veces el mensaje (es decir, luego de que presione 5 veces la tecla F10).

**Pista:** cuando se ejecuta la 5ta interrupción, deben deshabilitarse las mismas para evitar reaccionar a próximas pulsaciones de la tecla F10. Utilizar una variable global para registrar cuántas veces falta mostrar el mensaje antes de terminar.

5) **Timer, tres usos: periódico infinito, periódico finito, y única vez** ★★

- Escribir un programa que muestra el mensaje “Vamos las interrupciones!” una vez cada 2 segundos y no termine nunca.
- Modificar a) para que termine a los 10 segundos.
- Modificar a) para que solo imprima una vez, a los 10 segundos, y luego termine.
- Desafío:** Modificar a) para que el primer mensaje se imprima luego de 1 segundo, el segundo luego de 2 segundos, el tercero luego de 3, y así sucesivamente, hasta que se espere 255 en el último mensaje, y el programa termine,

6) **F10 con Timer**

- Conteo regresivo** ★★: Escribir un programa que implemente un conteo regresivo a partir de un valor (de 1 a 9) ingresado desde el teclado. El conteo debe comenzar al presionarse la tecla F10. El tiempo transcurrido debe mostrarse en pantalla, actualizándose el valor cada segundo. Por ejemplo, si se ingresa el valor 7, cuando se apreta F10 debe mostrarse en pantalla “7 6 5 4 3 2 1 0” en los 7 segundos siguientes

**Pista:** Una vez leído el dígito se puede activar solamente la interrupción F10, y cuando se invoque la subrutina F10 activar el Timer para que a partir de ese momento se muestre el conteo regresivo.





- Conteo regresivo con pausa** ★★ Escribir un programa que cuente de forma regresiva desde 9 hasta 0, mostrando el tiempo en pantalla.. Luego de que termine de contar, si se presiona F10 debe comenzar de nuevo la cuenta desde 9 hasta 0. Además, si en algún momento se presiona la tecla F10 durante el conteo, el mismo debe pausarse, y reanudarse sólo cuando se presiona nuevamente F10.

**Pista:** Utilizar un flag de estado “PAUSA” que se controla mediante la tecla F10, y que la subrutina del timer puede consultar para saber si debe decrementar el contador y mostrar en pantalla.

7) **Impresión con HANDSHAKE mediante interrupciones** ★★

- a) Escribir un programa que imprime "UNIVERSIDAD NACIONAL DE LA PLATA" en la impresora a través del HAND-SHAKE. El envío de los caracteres se realiza por **interrupciones** emitidas desde el HAND-SHAKE cada vez que detecta que la impresora se desocupa.
- b) Probar el programa utilizando:
  - 1) Una **velocidad de reloj** del 100%, y una **velocidad de impresión** del 25%
  - 2) Una **velocidad de reloj** del 25%, y una **velocidad de impresión** del 100%¿En qué caso sería mejor utilizar consulta de estado en lugar de interrupciones?
- c) Modificar a) de modo que el string a imprimir se lea desde teclado. El string a leer tiene una longitud fija de 10 caracteres, y se lee de forma completa *antes* de comenzar la impresión. Tener en cuenta entonces que las interrupciones deben esperar a que se ingrese todo el string.
- d) Modificar b) de modo que cuando se presione F10 se cancele la impresión. En tal caso, deben desactivarse las interrupciones del HANDSHAKE para evitar que se envíen más caracteres a imprimir.

## Parte 3: Ejercicios de repaso para el parcial

- 1) **Imprimir mensaje leído, longitud arbitraria** ★. Escribir un programa que solicite ingresar caracteres por teclado y los almacene en la memoria. Al presionar la tecla F10, se envían a la impresora a través de la PIO. No es necesario mostrar los caracteres en la pantalla.
- 2) **F10 y Luces** ★★ Escribir un programa que al presionar F10 encienda todas las luces, y al presionarlo nuevamente las apague. El programa nunca termina.
- 3) **Ruleta, F10 y azar** ★★ Escribir un programa que permita seleccionar un dígito al azar para jugar a la ruleta. Para eso, el programa principal debe iterar continuamente, cambiando un valor de un registro desde el '0' hasta el '9' (y volviendo al '0' luego del '9'). Cuando se presiona F10, la letra queda "seleccionada" en base al valor del registro, y debe mostrarse de inmediato en la pantalla. Luego el programa termina
- 4) **F10, lectura e Impresora con PIO** ★★★★★ Escribir un programa que debe mostrar en pantalla el mensaje "PRESIONE F10 PARA COMENZAR" y una vez que el usuario presiona F10, leer de teclado un mensaje de 10 caracteres. Este mensaje debe luego ser enviado a través de la pila a una subrutina para imprimirse en la impresora mediante el PIO. La configuración del PIO también debe hacerse en una subrutina aparte.
- 5) **Timer y Luces** ★★ Escribir un programa que debe leer el estado de las llaves y enviarlo a una subrutina. La subrutina debe llamarse **LUCES12** y recibe el estado por referencia a través de la pila. Si el estado indica que la llave 7 (la de más a la izquierda) está prendida entonces debe encender todas las luces por 12 segundos y luego apagar todas las luces. En caso contrario no debe hacer nada.
- 6) **F10, impresora con HS y dígitos** ★★★★★ Escribir un programa para imprimir en la impresora un conteo regresivo en base a un dígito ingresado por teclado. Por ejemplo, si el usuario ingresa "3", se debe imprimir "3 2 1 0". El programa comienza leyendo un carácter dígito con la subrutina  **LEER\_DIGITO**. Luego espera a que el usuario presione F10 para comenzar la impresión, y llama a la subrutina  **DESCENDER** que la implementa.  
 **LEER\_DIGITO** muestra en pantalla el mensaje "INGRESE UN NUMERO DEL 1 AL 9:" y lee un carácter de teclado. Si el carácter ingresado no corresponde al número solicitado, se debe volver a leer un carácter hasta que el usuario ingrese efectivamente un número del 1 al 9.  
 **DESCENDER** recibe el dígito por valor y por la pila, y envía a la impresora en forma descendente desde el número ingresado hasta el "0", utilizando el Handshake por Consulta de Estado.
- 7) **Dispositivo nuevo con PIO** ★★★★★ Escribir un programa para VonSim que envíe la cadena de caracteres "Hola!" a un dispositivo nuevo, conectado a los 8 bits del puerto PB. Este dispositivo recibe la cadena de a un carácter a la vez, esperando 5 segundos entre cada carácter. Para indicar al dispositivo que finalizó el envío de datos, se debe enviar el código ascii 0 luego de enviar todos los caracteres del string. El programa debe finalizar cuando se han enviado todos los caracteres de la cadena, o cuando se presiona la tecla F10, cancelando el envío cancelando el envío de los caracteres que restan.

**Ejemplo** para enviar la cadena "ASDF": Envío de la "A" → 5 segundos de espera → Envío de la "S" → 5 segundos de espera → Envío de la "D" → 5 segundos de espera → Envío de la "F" → 5 segundos de espera → Envío de 0