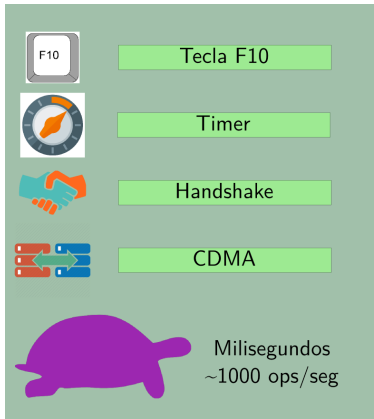


Interrupciones por Hardware en el simulador MSX88

Facundo Quiroga

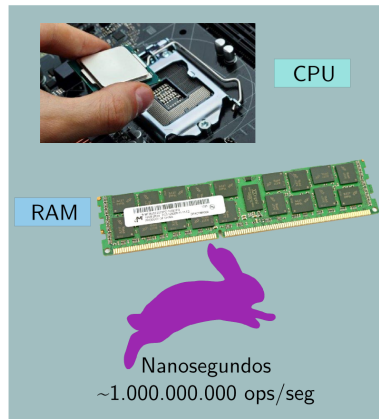
Velocidad de dispositivos vs cpu/memoria



A diagram on a green background comparing the speed of various devices to a CPU and memory. It lists four items with icons and labels in green boxes: 'Tecla F10' (F10 key icon), 'Timer' (timer icon), 'Handshake' (handshake icon), and 'CDMA' (CDMA icon). At the bottom, a purple turtle icon is labeled 'Milisegundos ~1000 ops/seg'.

- Tecla F10
- Timer
- Handshake
- CDMA

Milisegundos
~1000 ops/seg



A diagram on a blue background comparing the speed of various devices to a CPU and memory. It shows a CPU being installed in a socket (labeled 'CPU') and a RAM module (labeled 'RAM'). At the bottom, a purple rabbit icon is labeled 'Nanosegundos ~1.000.000.000 ops/seg'.

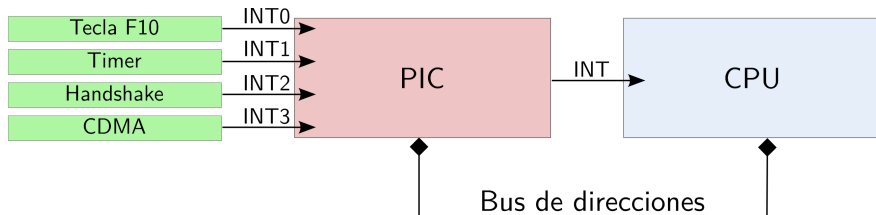
CPU

RAM

Nanosegundos
~1.000.000.000 ops/seg

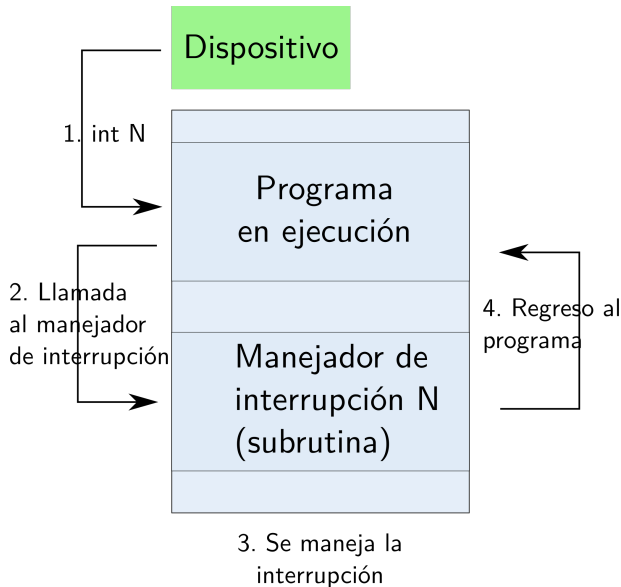
Los dispositivos deberían esperar a la CPU y no viceversa.

PIC: Conexion de los dispositivos



Los dispositivos interrumpen a la CPU a través del PIC

El PIC organiza la ejecución de una subrutina



La subrutina atiende la interrupción del dispositivo.

Escribiendo una subrutina de interrupción para la tecla F10

Cuando se presiona F10, se imprime un mensaje en pantalla.

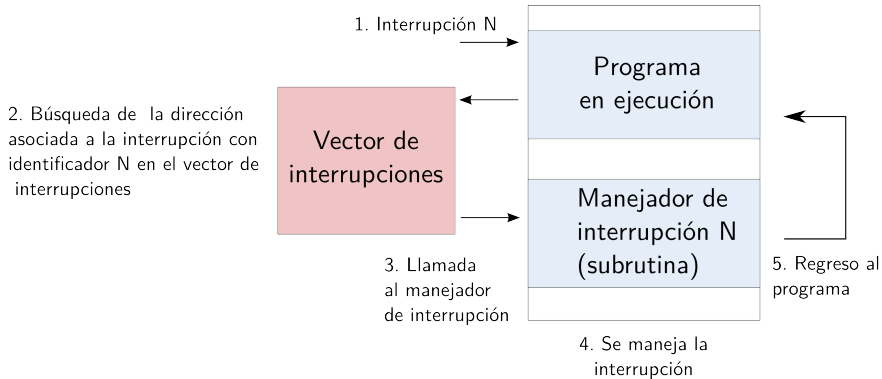
Subrutina especial llamada "manejador".

```
1          org 1000h
2 mensaje db "Has_presionado_la_tecla_F10!"
3 fin      db ?
4          org 3000h
5 rutf10: mov bx,offset mensaje
6          mov al, offset fin-offset mensaje
7          ; mostrar el mensaje en pantalla
8          int 7
9          ;avisar al PIC que termino la interrupcion (DESPUES)
10
11          ; IRET y no ret porque volvemos de una interrupcion
12          iret
13          org 2000h
14          ; configuracion del PIC (DESPUES)
15          ; configuracion del vector de interrupciones
16 loop: jmp loop ; bucle infinito
17          end
```

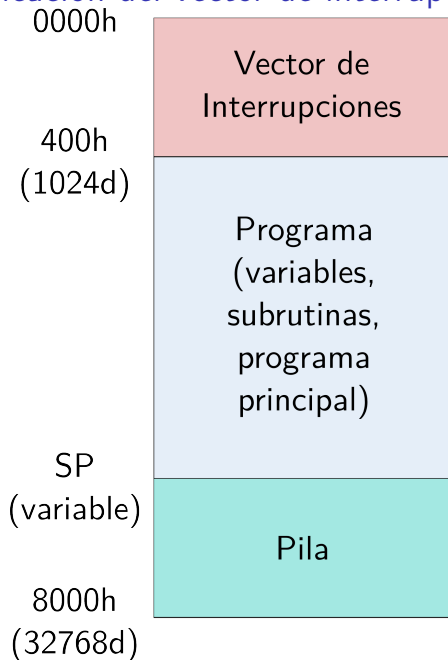
Pasos para usar el F10 mediante interrupciones por Hardware

1. Escribir la subrutina de atención. Es una subrutina normal salvo por que:
 - ▶ Finaliza con `iret` en lugar de `ret`.
 - ▶ Debe avisarle al PIC cuando termina (FALTA).
2. Configurar el PIC para que cuando el F10 interrumpa, se ejecute la subrutina de atención
 - ▶ Elegir un número de interrupción (cualquiera menos 0, 3, 6 o 7). Llamémoslo N (FALTA).
 - ▶ Poner la dirección de la subrutina en el vector de interrupciones, en el elemento N (FALTA).
 - ▶ Configurar el PIC, asignando N como número de interrupción del F10 (FALTA).
 - ▶ Configurar el PIC, habilitando las interrupciones del F10 (FALTA).

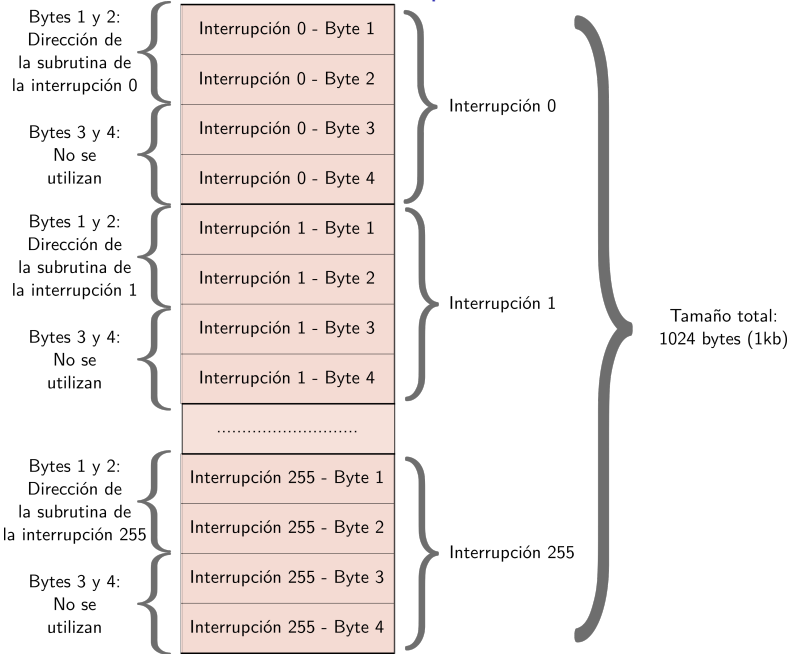
Cómo encontrar la subrutina: Vector de interrupciones



Ubicación del vector de interrupciones



Estructura del vector de interrupciones



Configurando el vector de interrupciones

Nosotros elegimos el número de interrupción. Digamos, el 12.

```
1      org 3000h  
2  rutf10: mov bx,offset mensaje  
3      .....  
4
```

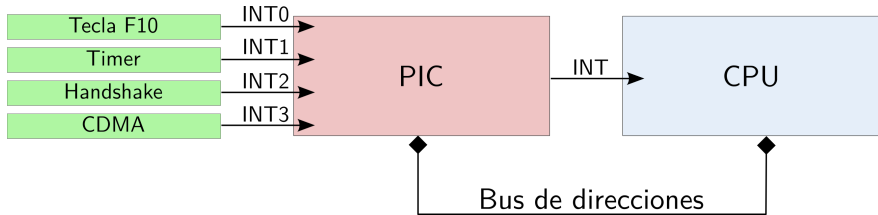
Forma 1 de configurarlo, en el programa principal:

```
1      mov bx, 48 ; 12*4  
2      mov [bx],3000h  
3
```

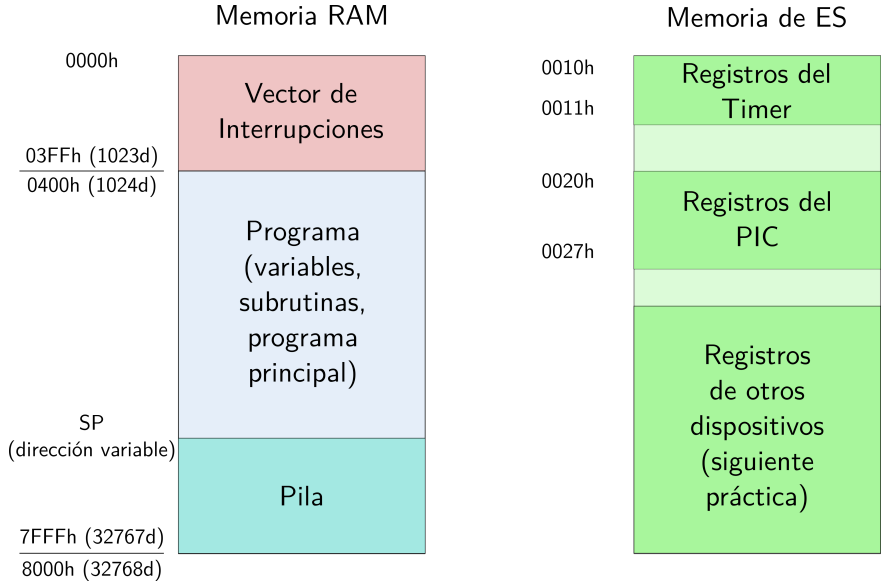
Forma 2 de configurarlo, en cualquier lado:

```
1      org 48; 12*4  
2  dir_rutf10 dw 3000h  
3
```

PIC: Conexion de los dispositivos



Configurando el PIC: memoria de entrada salida



Registros de entrada salida

- ▶ Cada dispositivo tiene uno, tienen direcciones.
- ▶ Se cambian con `out`.
- ▶ Se leen con `in`.
- ▶ Siempre con el registro `al` como intermediario.
- ▶ Si quiero ponerle el valor 5 al registro con dirección 20h, hago 2 cosas:
 - ▶ `mov al,5`.
 - ▶ `out 20h,al`
- ▶ Si quiero leer el valor del registro 20h, simplemente hago `in 20h,al`

Dirección	Registro	Nombre	Propósito	E/S
20h	EOI	Fin de interrupción	Avisa al PIC que se terminó una interrupción	S
21h	IMR	Máscara de interrupciones	Sus bits indican qué líneas de interrupción están habilitadas. Si el bit N vale 1, las interrupciones del dispositivo conectado a la línea INTN serán ignoradas. Si vale 0, las interrupciones del dispositivo serán atendidas en algún momento. Sólo importan los 4 bits menos significativos.	S
22h	IRR	Interrupciones pedidas	Sus bits indican qué dispositivos están solicitando una interrupción. Si el bit N vale 1, entonces el dispositivo conectado a la línea INTN está haciendo una solicitud. Sólo importan los 4 bits menos significativos.	E
23h	ISR	Interrupción en servicio	Sus bits indican si se está atendiendo la interrupción de algún dispositivo. Si el bit N vale 1, entonces el dispositivo conectado a la línea INTN está siendo atendido. Cómo en el MSX88 sólo se puede atender un dispositivo por vez, nunca habrá más de un bit del registro con el valor 1. Sólo importan los 4 bits menos significativos.	E
24h	INT0	ID de Línea INT0	Almacena el ID de la interrupción asociada al dispositivo F10 para buscar en el vector de interrupciones la dirección de comienzo de la subrutina que lo atiende.	S
25h	INT1	ID de Línea INT1	Almacena el ID de la interrupción asociada al dispositivo Timer para buscar en el vector de interrupciones la dirección de comienzo de la subrutina que lo atiende.	S
26h	INT2	ID de Línea INT2	Almacena el ID de la interrupción asociada al dispositivo Handshake para buscar en el vector de interrupciones la dirección de comienzo de la subrutina que lo atiende.	S
27h	INT3	ID de Línea INT3	Almacena el ID de la interrupción asociada al dispositivo CDMA para buscar en el vector de interrupciones la dirección de comienzo de la subrutina que lo atiende.	S

PIC: IMR, dirección 21h.

- ▶ Máscara de interrupciones.
- ▶ 8 bits, cada uno indica si el dispositivo está habilitando
- ▶ 0 para habilitado, 1 para deshabilitar

```
1      ; Habilito las interrupciones del dispositivo 0
2      ; Pongo la mascara 0111 1111
3      mov al,01111111b
4      out 21h,al
5
```

PIC: **IMR**, dirección 21h. Alternativa: definir la dirección como una constante.

```
1      IMR equ 21h
2
3      ....
4
5      ; Habilito las interrupciones del dispositivo 0
6      ; Pongo la mascara 0111 1111
7      mov al,01111111b
8      out IMR,al
9
```


PIC: **INT0**, dirección 24h, registro del F10.

- ▶ Asignarle el número de interrupción al dispositivo
- ▶ Poner el valor 12, que es el que elegimos antes.

```
1  INT0 equ 24h
2
3  ....
4
5  mov al,12
6  out INT0,al
7
```

PIC: **cli** y **sti**: activar o desactivar todas las interrupciones

- ▶ Mientras configuro el PIC, no quiero que estén habilitadas las interrupciones
- ▶ **cli** antes de configurar, **sti** luego de hacerlo.

```
1  cli  
2  ; configurar el PIC  
3  sti  
4
```

Configurando el PIC para recibir interrupciones de la tecla F10

```
1  EOI EQU 20h
2  IMR EQU 21h
3  INT0 EQU 24h
4
5  org 2000h
6  ; configuracion del PIC
7  cli ; desactivar todas las interrupciones
8  mov al,1111110b
9  out IMR,al; configuro el IMR
10 mov al, 12
11 out INT0, al; configuro como 12 el identificador del F10
12 sti ; activar todas las interrupciones
13
14 loop: jmp loop ; bucle infinito
15 end
```

PIC: **EOI**, dirección 20h

```
1      EOI equ 20h
2      ....
3      org 3000h
4      ; subrutina de atencion
5      .....
6
7      ; envio el valor 20h a la direccion de E/S 20h
8      mov al,20h
9      out EOI,al
10     iret
11
```

- OJO: la dirección y el valor a enviar son los mismos (20h)

Ejemplo completo

```
1      org 3000h
2  rutf10: mov bx, offset mensaje
3          mov al, offset fin - offset mensaje
4          int 7 ; mostrar el mensaje en pantalla
5          mov al, 20h
6          out EOI, al
7          iret
8
9  org 48; 12 * 4
10 dir_rutf10 dw 3000h; escribo sobre el vector de interrupciones
11
12 EOI EQU 20h
13 IMR EQU 21h
14 INT0 EQU 24h
15
16 org 2000h
17 cli ; desactivar todas las interrupciones
18 mov al, 1111110b
19 out IMR, al; configuro el IMR
20 mov al, 12
21 out INT0, al; configuro como 12 el identificador del F10
22 sti ; activar todas las interrupciones
23
24 loop: jmp loop ; bucle infinito
25 end
```

Esquema de un programa con interrupciones por hardware

Constantes: direcciones de registros
de ES y valores especiales como 20h

Poner la dirección del manejador
en el vector de interrupciones

Manejador de interrupción (subrutina)

Código de la subrutina

.....

Enviar 20h al registro EOI
iret

Programa Principal

cli

Configurar el registro IMR

Configurar el registro INTN

Configurar otros dispositivos

sti

Código del programa principal

.....

end

Pasos para usar el F10 mediante interrupciones por Hardware

1. Escribir la subrutina de atención. Es una subrutina normal salvo por que:
 - ▶ Finaliza con `iret` en lugar de `ret`.
 - ▶ Debe avisarle al PIC cuando termina enviando 20h al `EOI`.
2. Configurar el PIC para que cuando el F10 interrumpa, se ejecute la subrutina de atención.
 - ▶ Elegir un número de interrupción (cualquiera menos 0, 3, 6 o 7). Llamémoslo N.
 - ▶ Poner la dirección de la subrutina en el vector de interrupciones, en el elemento N.
 - ▶ Configurar el PIC, poniendo el valor N en el `INT0`.
 - ▶ Configurar el PIC, poniendo el valor 0111 1111 en el `IMR`.
 - ▶ La configuración se hace entre instrucciones `cli` y `sti`.