

Conceptos de Arquitectura de Computadoras

Clase 2

Interrupciones

Interrupciones

- Mecanismo mediante el cual se puede cambiar el procesamiento normal de la CPU.
 - Ejecución secuencial de instrucciones de un programa
- Pueden ser de origen interno o externo a la CPU.

¿Porqué Interrumpir?

- Por resultado de una ejecución de una instrucción
 - Ej: desbordamiento aritmético ("overflow"), división por cero
- Por un temporizador interno del procesador.
 - Permite al S.O. realizar ciertas funciones de manera regular.
- Por una operación de E/S.
 - Ej: para indicar la finalización normal de una operación.
- Por un fallo de hardware.
 - Ej: error de paridad en la memoria, pérdida de energía.

¿Qué hacer si interrumpen?

- En casi todos los casos, implica transferir el control a otro programa (el GESTOR).
 - antes de ello hay salvar el “estado del procesador”
- La función del Gestor es:
 - corregir (o responder a) la causa que ocasionó la solicitud de interrupción
 - guardar/restaurar si usa/modifica valores en registros
 - retornar a la ejecución normal del programa interrumpido
- ¿continúo ó repito la instrucción interrumpida?

Tipos de interrupciones

Se pueden considerar

- No Enmascarables: las que NO pueden ignorarse
 - Indican eventos peligrosos o de alta prioridad.
 - Requieren respuesta eficiente y rápida
- Enmascarables: pueden ser ignoradas.
 - Sus eventos no configuran peligro ó pueden esperar
 - La posible solicitud de interrupción puede inhibirse con instrucciones especiales.

Interrupciones por hardware

- Son las generadas por dispositivos de E/S.
- Son las “verdaderas” interrupciones.
- El sistema de cómputo tiene que manejar estos eventos externos “no planeados” ó “asincrónicos”.
- No están relacionadas con el proceso en ejecución en ese momento.
- Son conocidas como *interrupt request*.

Traps/excepciones

- Interrupciones por hardware creadas por el procesador moderno en respuesta a ciertos eventos internos:
 - Condiciones excepcionales: overflow en ALU de punto flotante.
 - Falla de programa: tratar de ejecutar una instrucción no definida.
 - Fallas de hardware: error de paridad de memoria.
 - Accesos no alineados ó a zonas de memoria protegidos

Interrupciones por software

Son instrucciones explícitas que afectan al procesador de la misma manera que las interrupciones por hardware.

- Permiten depurar los gestores de interrupción.
- Pueden ser usadas para invocar funciones del S.O.
 - permite que las subrutinas del S.O. se carguen en 'algún' lugar y puedan utilizarse.
- No requieren conocer la dirección de la rutina en tiempo de ejecución.

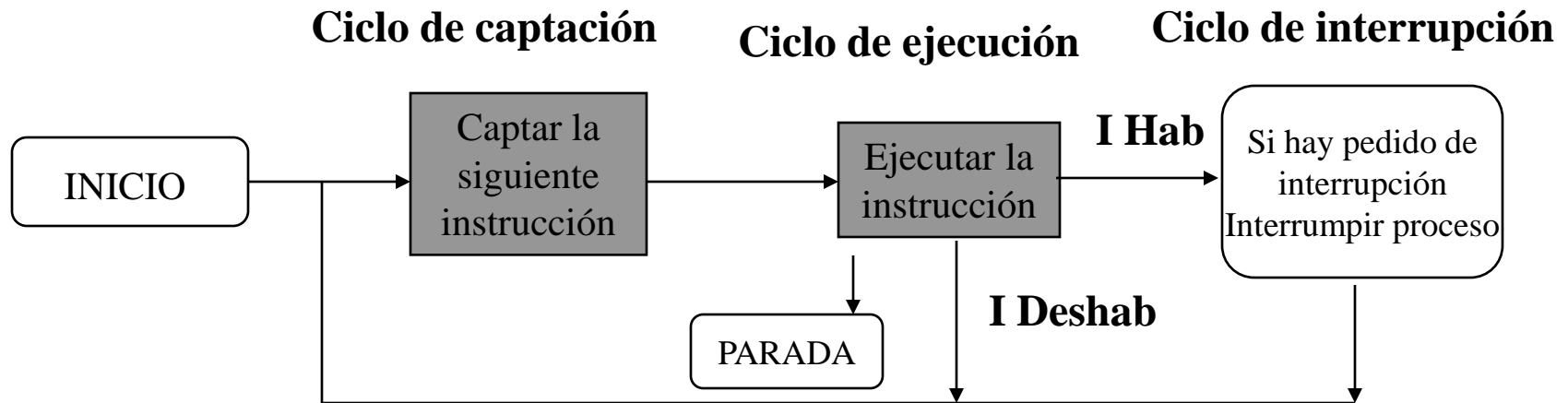
Interrupciones por software (2)

Normalmente no se permiten hacer una llamada directa a una dirección de una función del S.O., por estar en una zona reservada y protegida por derechos de autor del S.O.

- ¿Qué pasa si no tuviera las interrupciones por software?
 - Debería escribir todas las funciones que necesito ó
 - Al cargar un programa habría que “mirar” todas las llamadas a funciones del BIOS y S.O. y reemplazar en el código las direcciones de todas estas funciones invocadas.

Ciclo de instrucción

- Tres pasos:
 - Captación
 - Ejecución
 - Gestión de interrupciones



Ciclo de interrupción

- Se comprueba si se ha solicitado alguna interrupción.
 - indicada por una señal (flag) de pedido de interrupción.
- Si no hay señal se capta la siguiente instrucción.
- Si hay algún pedido de interrupción pendiente:
 - Se suspende la ejecución del programa en curso
 - Guarda su contexto (próxima instrucción a ejecutar y el estado del procesador)
 - Carga el PC con la dirección de comienzo del GESTOR (rutina de gestión de interrupción). Se inhiben otros pedidos de interrupción.
 - Finalizada la rutina de gestión, el procesador retoma la ejecución del programa del usuario en el punto de interrupción.

¿cómo trabajan?

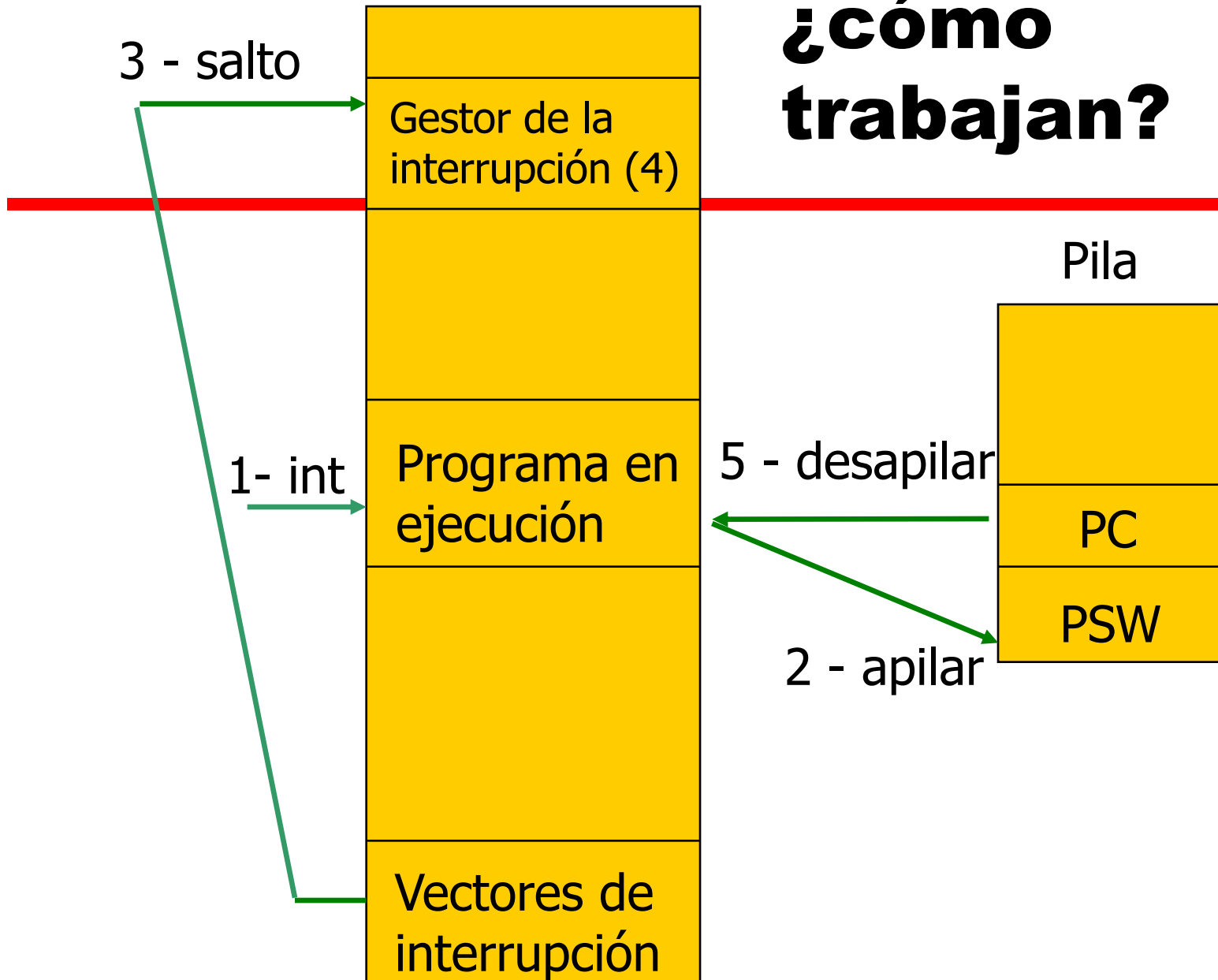
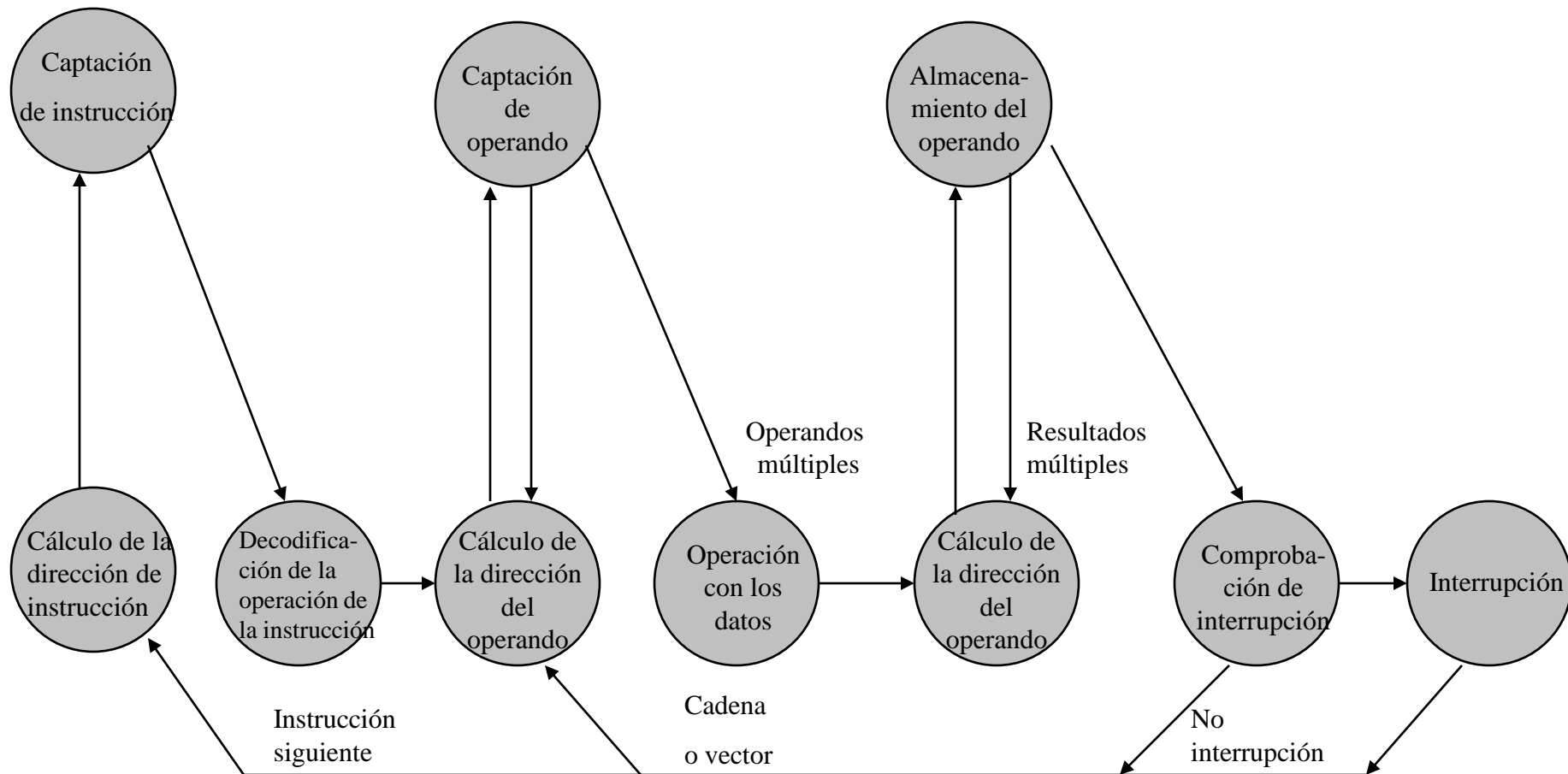


Diagrama de estados de un ciclo de instrucción con interrupciones



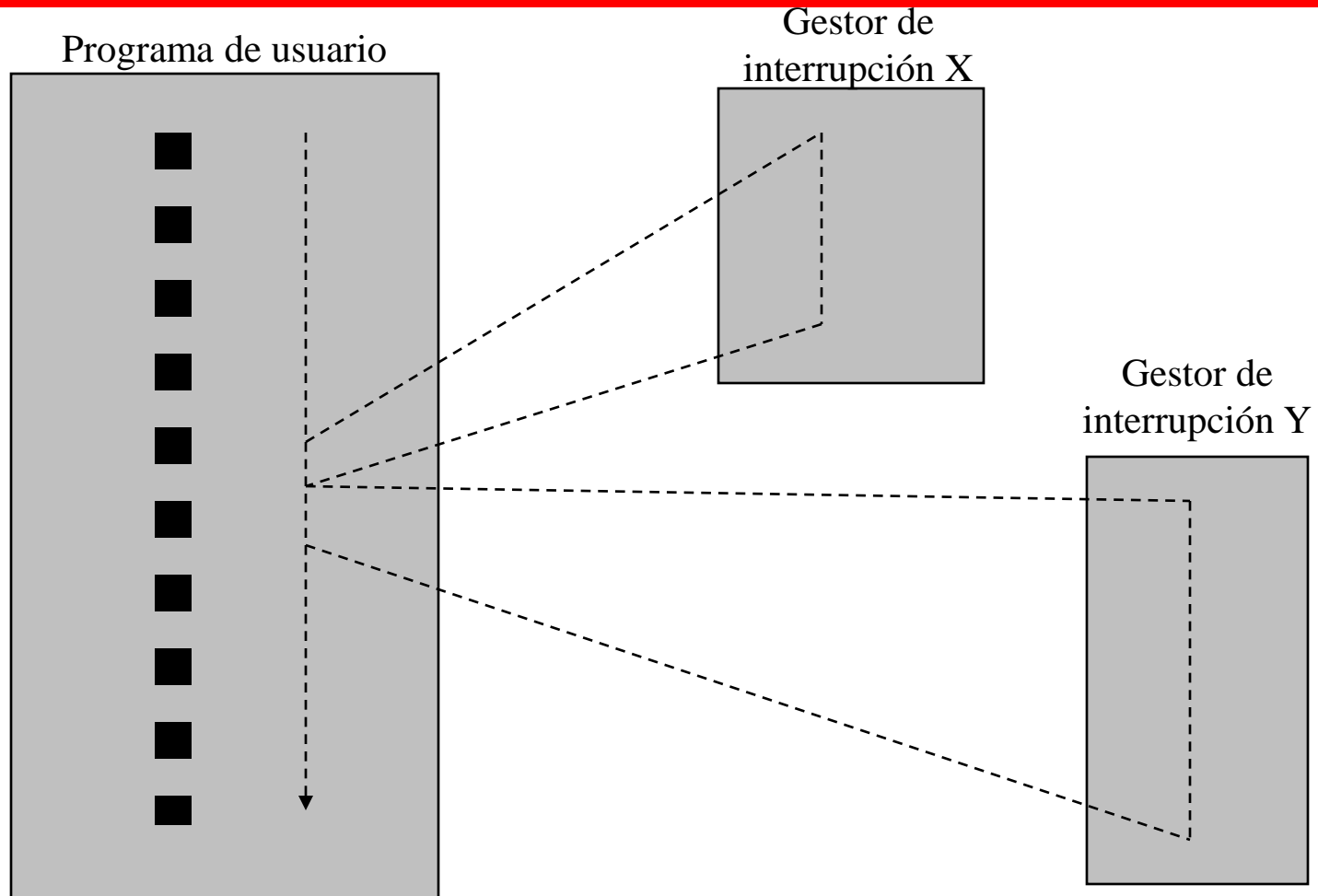
Interrupciones múltiples (1)

- Con Interrupciones inhabilitadas
 - El procesador puede y debe ignorar la señal de petición de interrupción si se produce una interrupción en ese momento.
 - Si se hubiera generado una interrupción se mantiene pendiente y se examinará luego una vez que se hayan habilitado nuevamente.
- Con Interrupciones habilitadas
 - Ante una solicitud, se inhabilitan
 - Se gestiona la misma y luego se habilitan otra vez.
- Por lo tanto las interrupciones se manejan en un orden secuencial estricto.

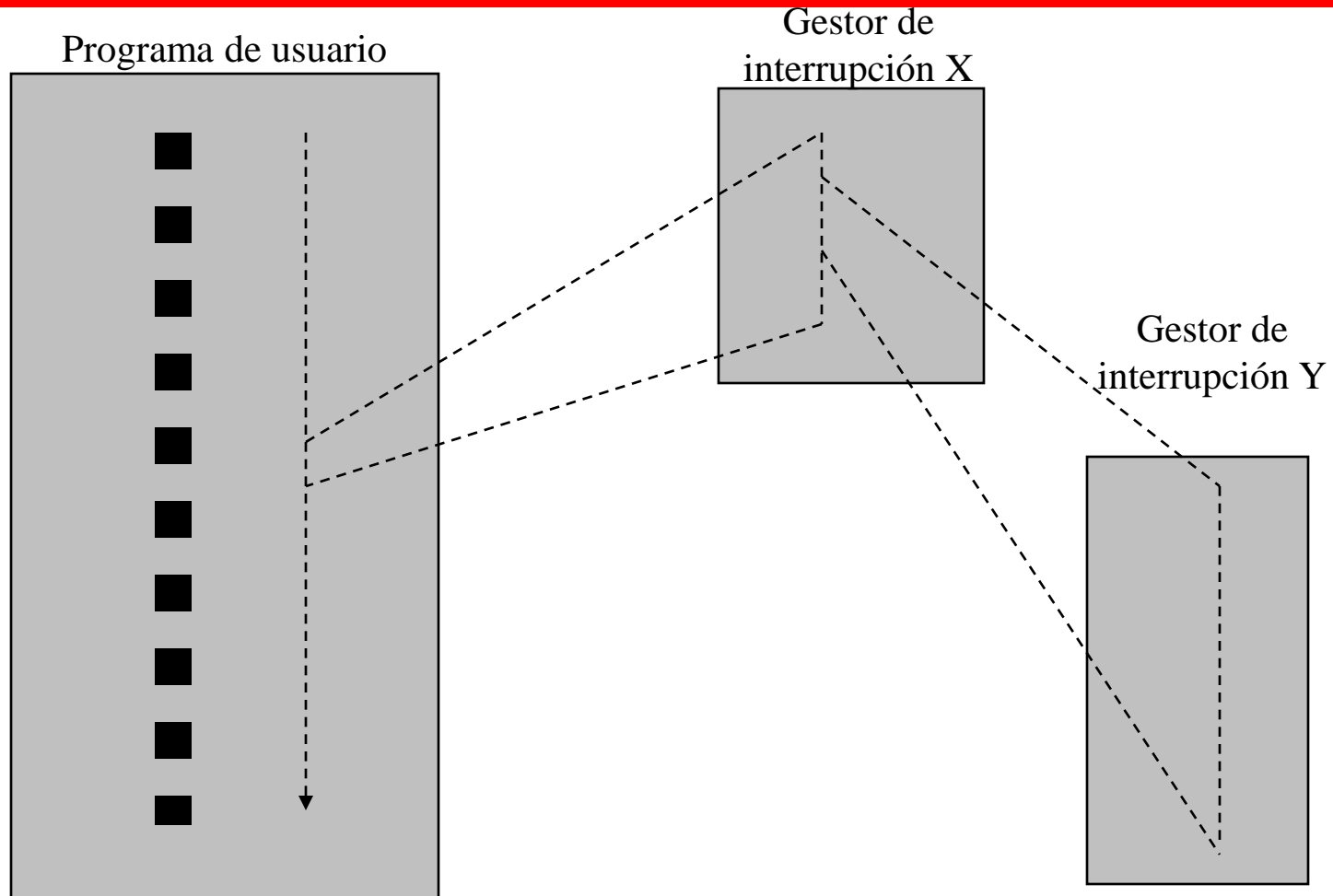
Interrupciones múltiples (2)

- Con prioridades
 - Una interrupción de prioridad más alta puede interrumpir a un gestor de interrupción de prioridad menor.
 - Cuando se ha gestionado la interrupción de prioridad más alta, el procesador vuelve a las interrupciones previas (de menor prioridad).
 - Terminadas todas las rutinas de gestión de interrupciones se retoma el programa del usuario.
- Las interrupciones se manejan anidando gestores.

Procesamiento de interrupciones secuenciales



Procesamiento de interrupciones anidadas (priorización)



Reconocimiento de interrupciones

Interrupciones multinivel

- Cada dispositivo que puede provocar interrupción tiene una entrada física de interrupción conectada a la CPU.
- Es muy sencillo, pero muy caro.

Línea de interrupción única

- Una sola entrada física de pedido de interrupción a la que están conectados todos los dispositivos.
- Se debe “preguntar” a cada dispositivo si ha producido el pedido de interrupción (técnica Polling/encuesta).

Reconocimiento de interrupciones (2)

Interrupciones vectorizadas

- El dispositivo que quiere interrumpir además de la señal de pedido de interrupción, debe colocar en el bus de datos un identificador (vector).
 - Lo coloca el periférico directamente ó
 - Controlador de Interrupciones (que se ocupa de todo).

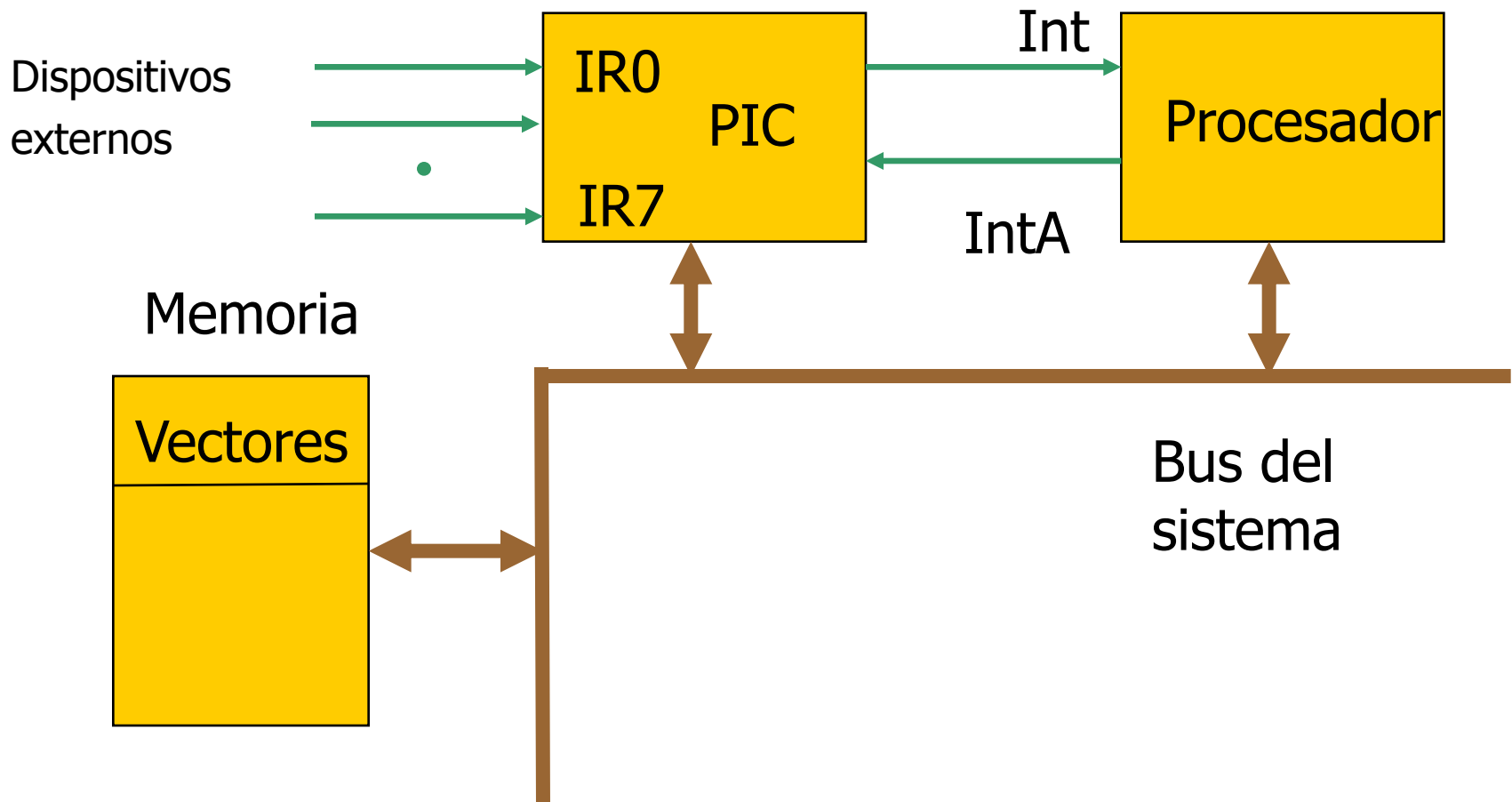
Escenario de trabajo

- Si el procesador tiene una única entrada de pedido de interrupciones.
- Si tenemos varios productores de interrupciones

¿Cómo lo solucionamos?

- Dispositivo controlador programable de interrupciones **'PIC'**.

Conexionado



Interrupciones del MSX88

- Hardware

- Línea INT

- Con respuesta de reconocimiento INTA

- Línea NMI

Procesos de atención son por salto indirecto

- Software

- Instrucción INT xx

Para retorno desde el gestor debe usarse la instrucción IRET

- Proceso de atención vectorizado

Tabla de vectores de Interrupción

- Es el nexo entre tipo de interrupción (0...255) y el procedimiento designado para atenderla.
- Cada entrada es una doble palabra (4 bytes).
 - Dirección del procedimiento que brinda el servicio.
 - Ej: 0000yyyy, donde yyyy es la dirección lógica/física.
- Vectores preasignados
 - Tipo 0 – finaliza ejecución de programa
 - Tipo 3 – punto de parada para depuración/seguimiento
 - Tipo 6 – lectura de entrada std. Requiere el uso de BX.
 - Tipo 7 – escritura de salida std. Requiere BX y AL.

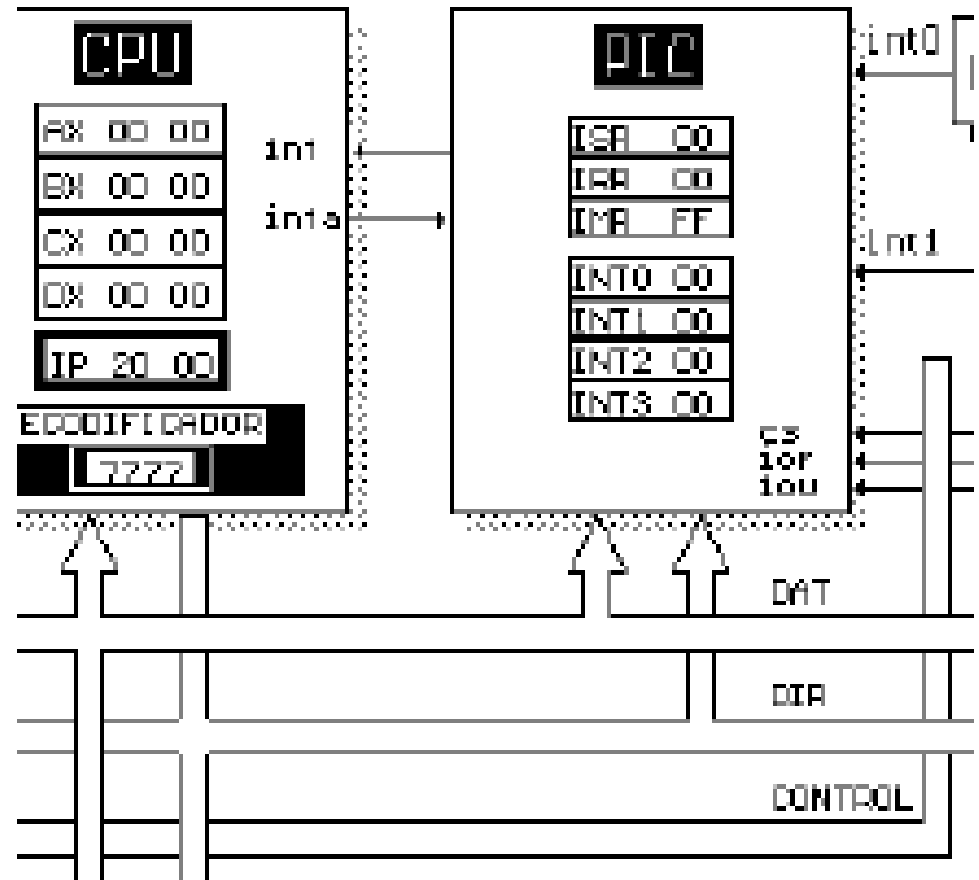
Tipo6 --> asume
que va a haber
un ingreso de
datos de teclado
Usa bx para
guardar el valor
de ASCII que me
manda mediante
ingreso de datos

tipo 7 --> saca el dato que esta en memoria accediendo a la direc q esta en bx, y lo almaceno en AL

Controlador de Interrupciones

Registros internos PIC

- EOI: para comandos
 - Para fin de int escribir 20H
- IMR: máscara de int
 - enmascara con '1'
- IRR: petición de int
 - Indica con bit en 1
- ISR: int en servicio
 - Indica con bit en 1
- INT0...INT7
 - c/u con su vector



Conexionado y direccionamiento

- Los registros internos del PIC se sitúan a partir de la dirección 20H.
- Son accedidos con operaciones lectura y escritura en el espacio de E/S (IN y OUT).
- Interrupciones hardware asignadas
 - INT0 – tecla F10
 - INT1 – Timer
 - INT2 – Handshake
 - INT3 – DMA
 - INT4 a INT7 no usadas

Referencias

- William Stallings, Capítulo 3.
- MSX88, Manual de usuario.

Lecturas recomendadas

- “Interrupciones en la arquitectura INTEL IA32”, Blázquez, J.M. 2004.