

*RESETS*  
&  
*INTERRUPCIONES*  
  
*EN EL CPU08*

*Curso de Microcontroladores*

*Familia HC908 Flash de Motorola*

*Parte II*

ING. DANIEL DI LELLA DDFAE For Motorola Products

# *Resets & Interrupciones*

Los Resets inicializan al CPU a un estado conocido !!!!.

Todos los resets son manejados por medio del **System Integration Module (S.I.M )**

- **Tipos de Resets:**

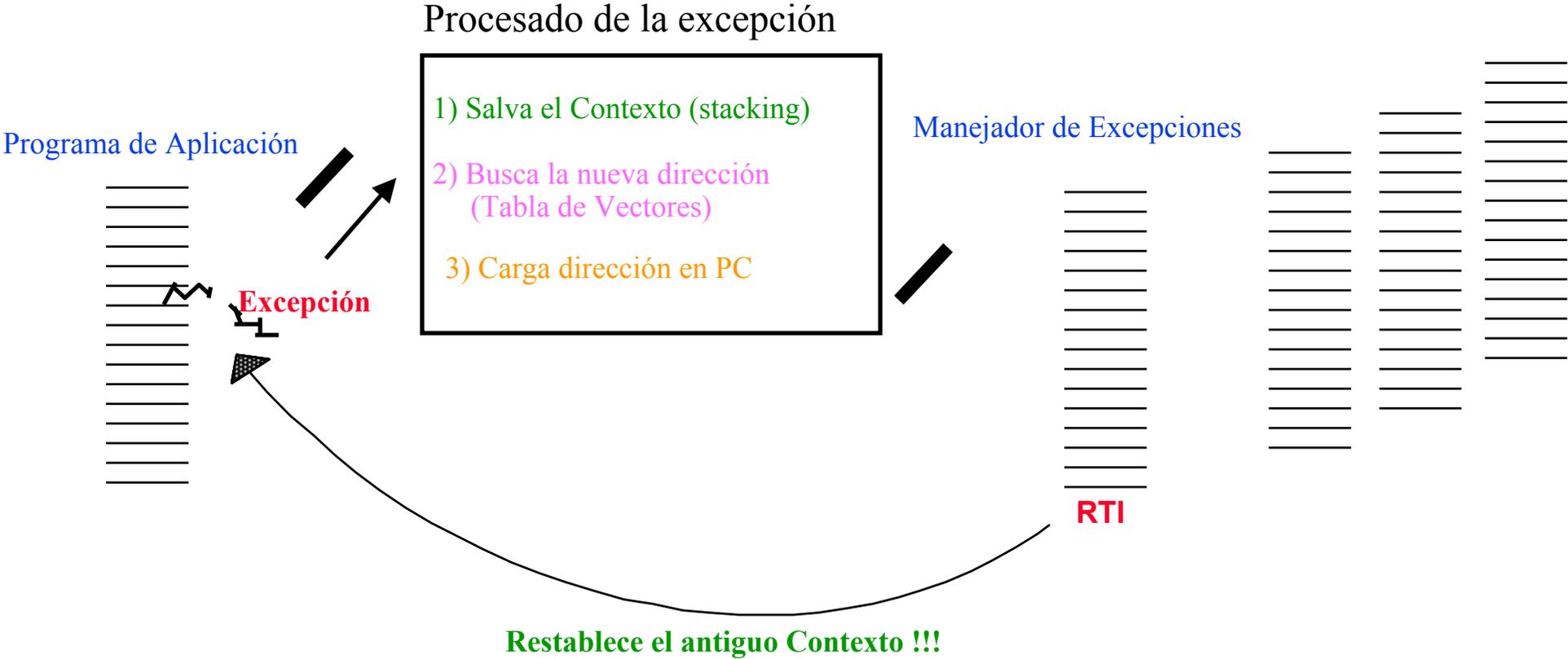
- Power On
- COP
- Illegal Address
- External Pin
- LVI
- Illegal Opcode

Las Interrupciones suspenden el procesamiento normal para que el CPU pueda realizar algunos servicios requeridos !!!!.

- **Fuentes de Interrupciones:**

- IRQ external PIN
- SCI
- TIM
- CGM (PLL)
- IRQ / KBI
- SPI
- ADC
- SWI

# Procesando Excepciones



## *Procesando Excepciones*

**Resets e Interrupciones** son excepciones del CPU (al proceso normal) , o sea “rompen” con la secuencialidad normal de la ejecución de un programa.

Determinando que tipo de manejo se requiere, se llama al proceso de excepción.

El procesamiento de la excepción es manejado a traves de tareas discretas.

- Diferencias para el reconocimiento en el resets y las interrupciones
  - Arbitraje
  - Stacking (apilado)
  - Búsqueda de vectores.

## *Procesando Excepciones - Reconocimiento -*

### **Detección de Reset o Interrupciones pendientes**

#### **Resets:**

- Es reconocida y ejecutada **inmediatamente** una vez ingresada (no hay sincronismo, es una excepción “imperativa”).

#### **Interrupciones:**

- Es reconocida durante el último ciclo de la instrucción corriente.
  - Al menos que ingrese durante el último ciclo
    - Entonces será reconocida durante el último ciclo de la próxima instrucción.
- Actúa después del último ciclo de la instrucción corriente.
- Hay sincronismo.

## *Procesando Excepciones - Arbitraje -*

### **Resets:**

Determinación de igual y más alta prioridad

- No arbitraje !!

### **Interrupciones:**

- Diferiendo prioridades
  - Mas bajas que los resets
- Realizado por el **SIM**

## Procesando Excepciones - Stacking (apilado) -

Salvando la información del CPU....

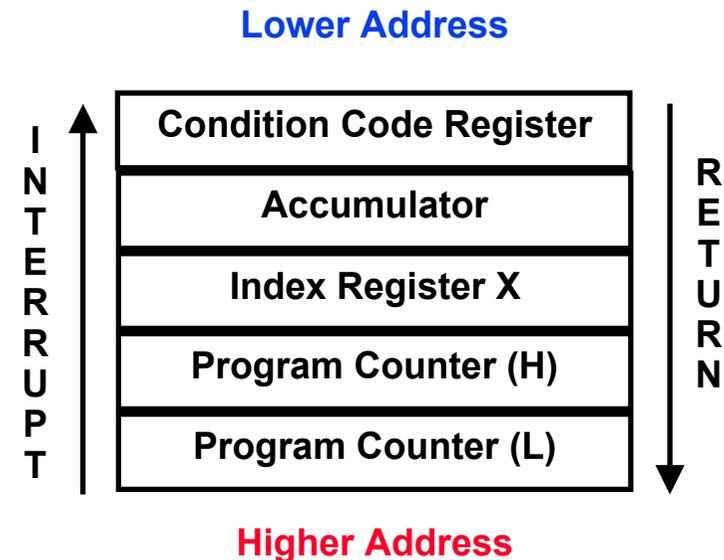
### Resets:

- No se realiza stacking (apilado), ya que **no es necesario** salvar registros.
- Se resetea el estado del CPU (a estados conocidos)

### Interrupciones:

- Stacks (apila) registros del CPU
  - **PC, X, A, CCR**
- El registro **“H”** **no es apilado** para mantener compatibilidad con HC705 !!. Usar instrucciones **PULL H & PUSH H** para ello.

### MAPA DEL STACK EN RAM (Registros del CPU)



# Procesando Excepciones - Búsqueda de Vectores -

## RESETS:

Todas usan el mismo vector (igual dirección)

- Puede determinarse la fuente del Reset examinando un registro del **SIM** ( **SIM Reset Status Register** ) **SRSR**

## Interrupciones:

- El Vector depende de la fuente de interrupción !!!

68HC908GP32 Vector Table

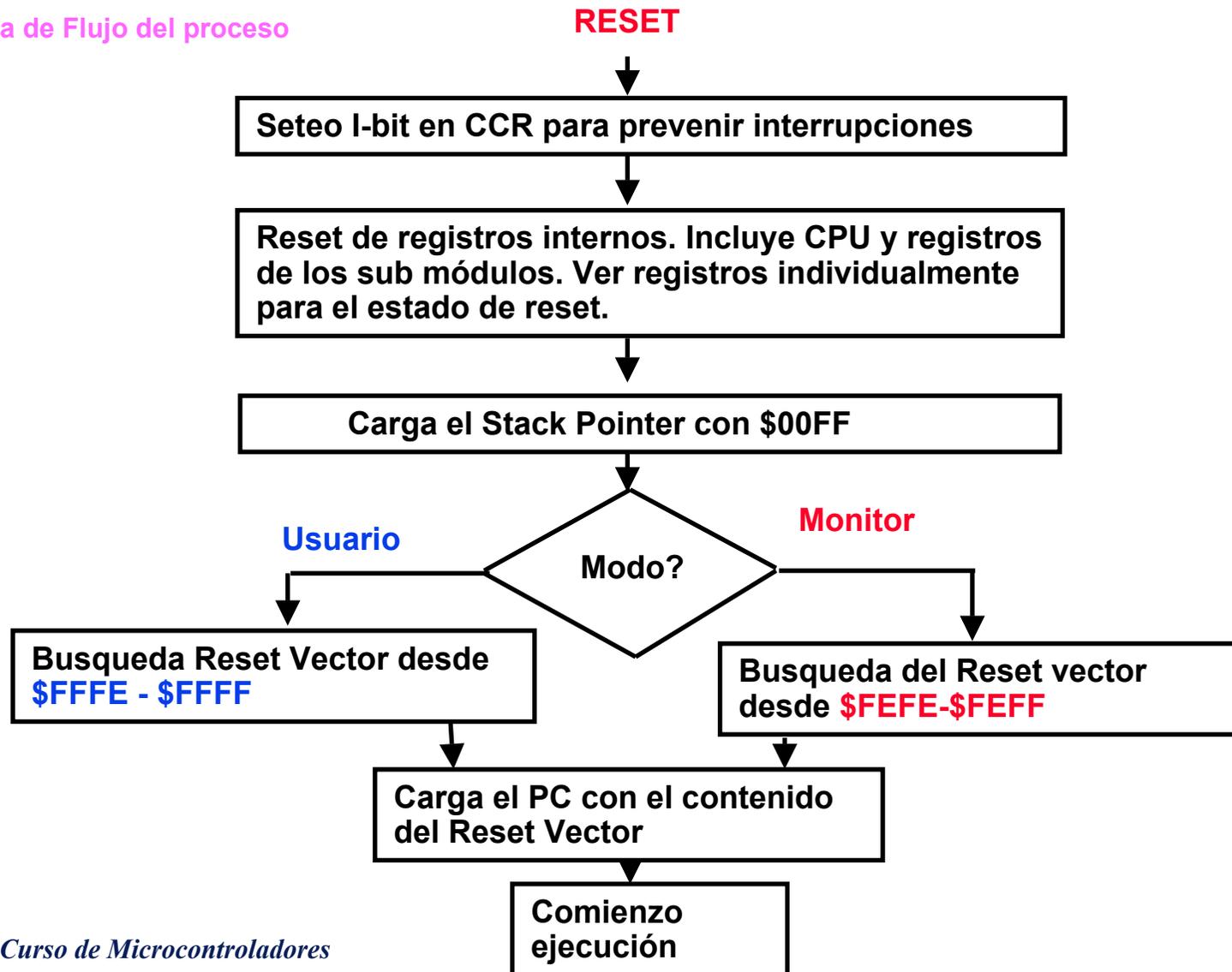
\$FFDC - \$FFDD	Timebase
\$FFDE - \$FFDF	ADC conv. complete
\$FFE0 - \$FFE1	Keyboard pins
\$FFE2 - \$FFE3	SCI Tx Empty /complete
\$FFE4 - \$FFE5	SCI Rx full/idle
\$FFE6 - \$FFE7	SCI Errors
\$FFE8 - \$FFE9	SPI Tx Empty
\$FFEA - \$FFEB	SPI Rx full/over/fault
\$FFEC - \$FFED	TIM2 Overflow
\$FFEE - \$FFEF	TIM2 Channel 1
\$FFF0 - \$FFF1	TIM2 Channel 0
\$FFF2 - \$FFF3	TIM1 overflow
\$FFF4 - \$FFF5	TIM1 Channel 1
\$FFF6 - \$FFF7	TIM1 Channel 0
\$FFF8 - \$FFF9	CGM (PLL)
\$FFFA - \$FFFB	IRQ pin
\$FFFC - \$FFFD	SWI Instruction
\$FFFE - \$FFFF	RESET

Más baja  
prioridad !!!

Más alta  
prioridad !!!

# Procesando el RESET

Diagrama de Flujo del proceso



Curso de Microcontroladores

Familia HC908 Flash de Motorola

Parte II

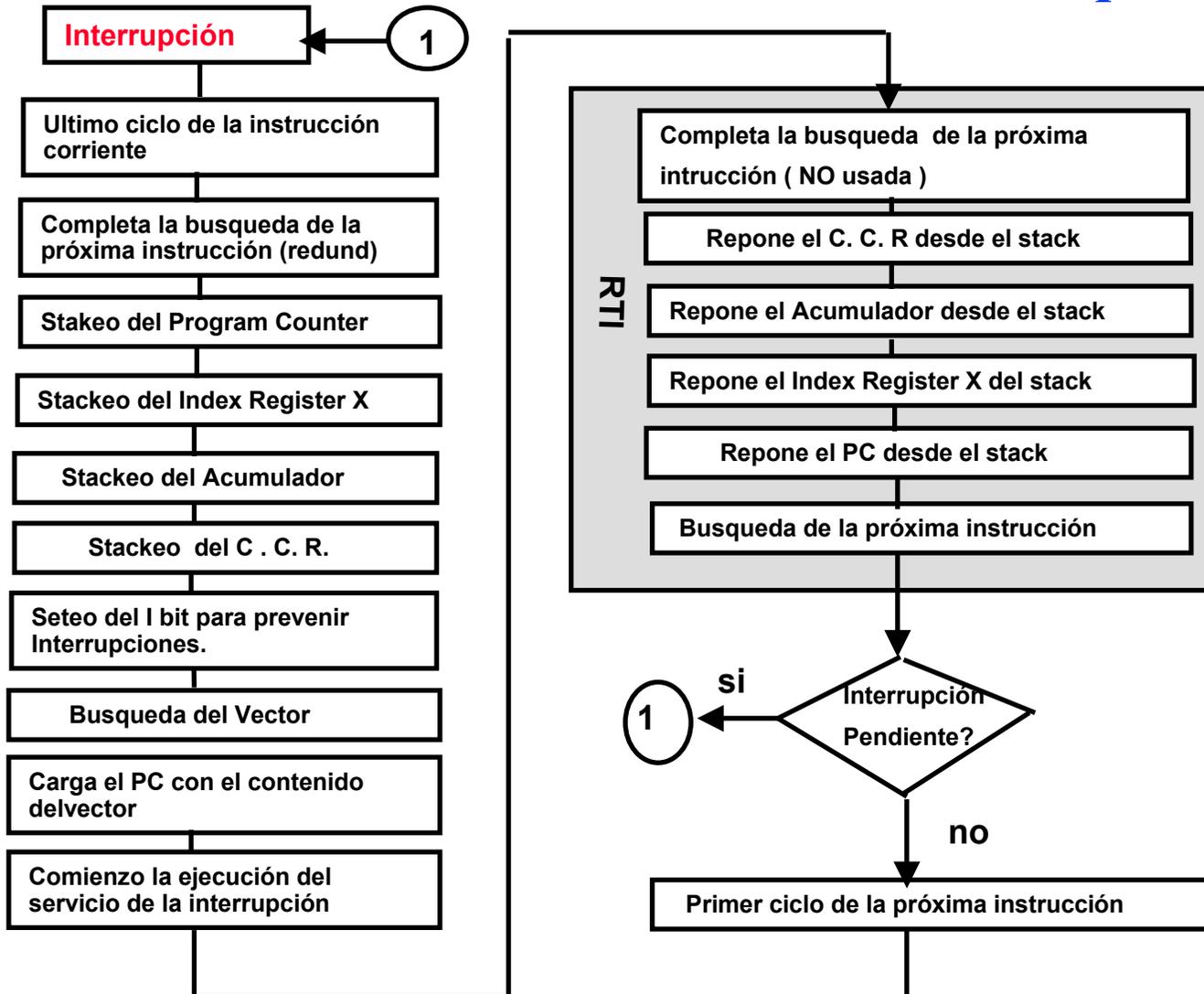
ING. DANIEL DI LELLA DDFAE For Motorola Products

## Procesando la Interrupción

Diagrama de Flujo del proceso

**Nota:** Index register H no se salva en el stack en forma automática .

El **I-bit** se limpia por la reposición del C.C.R. Desde el stack (al ejecutar la instrucción RTI)



Curso de Microcontroladores

Familia HC908 Flash de Motorola

Parte II

ING. DANIEL DI LELLA DDFAE For Motorola Products

## *Enmascaramiento (Masking)*

Habilitación / Deshabilitación del procesamiento de excepciones.

### Resets:

- **NO puede ser enmascarado (No puede ser impedido!!!.)**

### Interrupciones:

- Pueden ser enmascaradas (pueden ser impedidas !!!)
- **1 bit** habilita/deshabilita **TODAS** las interrupciones a procesar
- **Mascaras Locales** en periféricos que permiten **enmascarar individualmente interrupciones.**
- **Disparo de Interrupciones Externas por Flanco o por Flanco & Nivel.**

**Fin Capítulo 6 !!**