



# Sistemas Electrónicos Digitales

## Tema #3

### 4. Interrupciones

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70



# Temario

1. Introducción
2. GPIO: General Purpose Input/Output
3. Arquitectura Arm Cortex-M4
4. **Interrupciones**
5. C en ensamblador
6. Temporizadores (Timers)
7. Direct Memory Access
8. Comunicaciones Serie
9. Conversores A/D y D/A

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70



# INDICE ESPECÍFICO

- **Conceptos de Excepción e Interrupción**
  - Entrando en una Excepción
  - Saliendo de una Excepción
- **El núcleo de las Interrupciones**
- **Diseño de programa con Interrupciones**
  - Compartiendo datos de forma segura entre ISRs y otros hilos.

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70



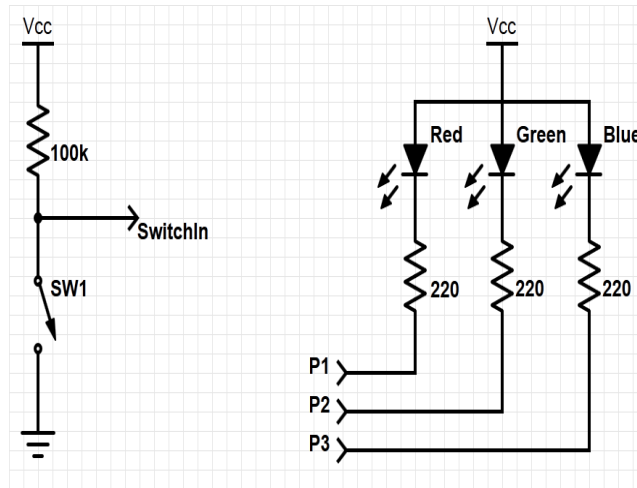
# INDICE ESPECÍFICO

- **Conceptos de Excepción e Interrupción**
  - Entrando en una Excepción
  - Saliendo de una Excepción
- **El núcleo de las Interrupciones**
- **Diseño de programa con Interrupciones**
  - Compartiendo datos de forma segura entre ISRs y otros hilos.

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# Estudio mediante un ejemplo



- Objetivo del ejemplo: cambiar el color del LED RGB de la placa de desarrollo cuando se presiona el pulsador sw1

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# ¿Cómo detectar que el pulsador se ha presionado?

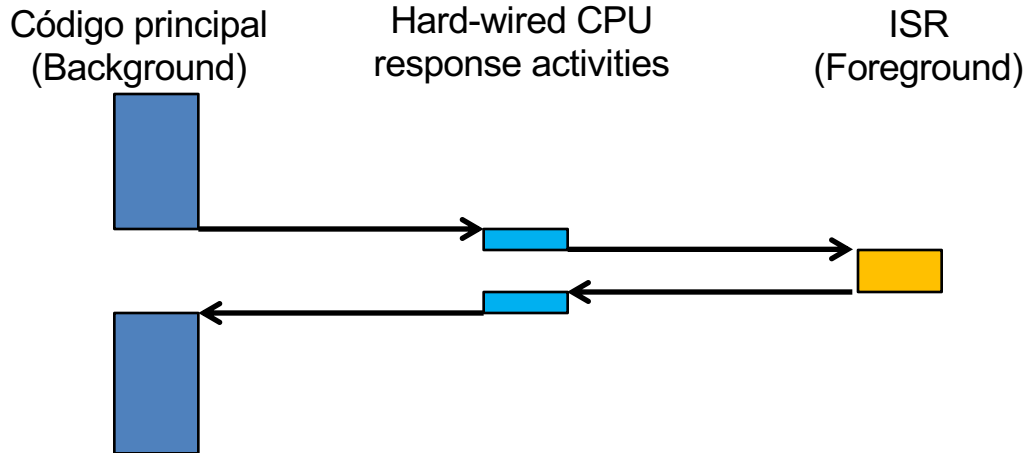
- **Polling** – comprobar en el bucle principal el estado del botón
  - **Lento**: es necesario comprobar explícitamente si el botón ha sido apretado.
  - **Desperdicio de tiempo de CPU**: Cuanto mas rápido necesitemos la respuesta más rápido tenemos que chequear el estado.
  - **Dificultad de escalado**: Es difícil construir un Sistema que tenga muchas actividades que tengan que ser atendidas con rapidez. El tiempo de respuesta depende del tiempo perdido en el resto de las tareas.
- **Interrupciones** – se usa hardware específico en el microcontrolador para detectar eventos y ejecutar código específico (**Interrupt Service Routine - ISR**) como respuesta
  - **Eficiente**: el código se ejecuta solo cuando es necesario
  - **Rápido**: es un mecanismo realizado por hardware.

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# Secuencia de procesamiento de una interrupción

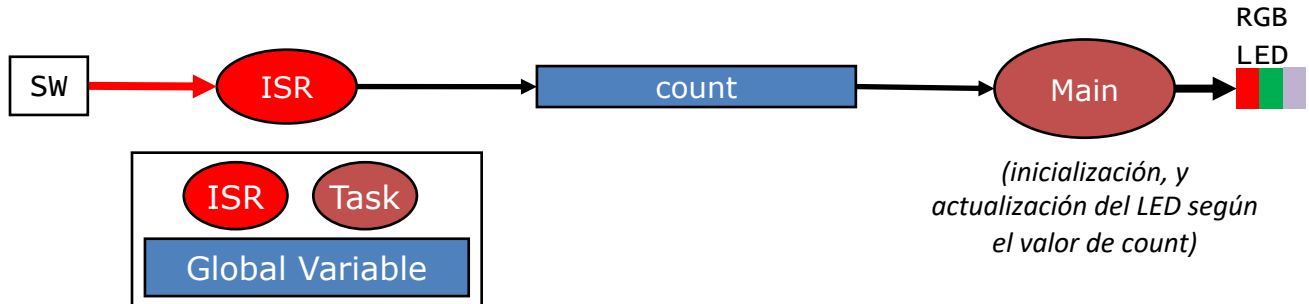


- El código principal (background) se está ejecutando
- Se dispara la interrupción
- El procesador realiza algunas tareas “hard-wired” (digamos que son automáticas)

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

## Ejemplo: requisitos y diseño



- Req1: Cuando se presiona el botón (SW), ISR incrementará la variable de cuenta (*count*)
- Req2: el código principal iluminará los LED de acuerdo con el valor de *count* en secuencia binaria (Azul: 4, Verde: 2, Rojo: 1)
- Req3: ISR tendrá prioridad (sobre el código principal) cuando se esté ejecutando

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70



# Procesado de una interrupción

- Vamos a estudiar el comportamiento del procesador ante una excepción

```
main.c
4
5 static int count = 0;
6
7 void button_press_isr(int sources) {
8     gpio_set(P_DBG_ISR, 1);
9     if (sources & (1 << GET_PIN_INDEX(P_SW))) {
10         count++;
11     }
12     gpio_set(P_DBG_ISR, 0);
13 }
14
15 int main(void) {
16     // Initialise LEDs.
17     leds_init();
```

Cartagena99

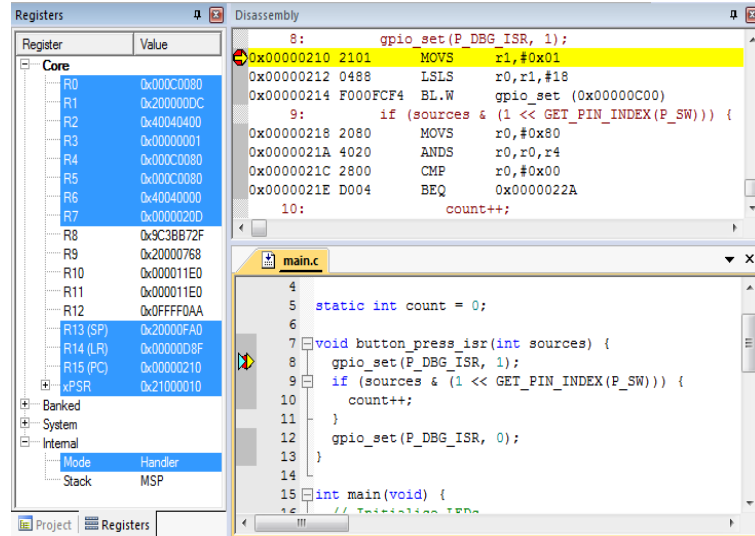
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# Uso del Debugger para el estudio

- Esta herramienta permite ver los valores de los registros, la pila, el código fuente y el código objeto (traducción del fuente a lenguaje ensamblador)

- Notas:

- El compilador genera código para la entrada a la función de excepción.
- Se ha de colocar el “breakpoint” en la



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

Cartagena99



# INDICE ESPECÍFICO

- **Conceptos de Excepción e Interrupción**
  - **Entrando en una Excepción**
  - Saliendo de una Excepción
- **El núcleo de las Interrupciones**
- **Diseño de programa con Interrupciones**
  - Compartiendo datos de forma segura entre ISRs y otros hilos.

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# Entrando en una Interrupción

## Secuencia del proceso de entrada de una interrupción:

1. Terminar la instrucción actual (excepto para instrucciones largas)
2. Salvar el contexto (8 registros de 32-bit) en la pila actual (MSP o PSP):
  - xPSR, Dirección de retorno, LR (R14), R12, R3, R2, R1, R0
3. Cambiar a modo privilegiado, usar MSP
4. Cargar el PC con la dirección de la rutina de excepción
5. Cargar LR con el valor EXC\_RETURN
6. Cargar IPSR con el número de excepción
7. Empezar la ejecución de la rutina de excepción

Usualmente, este proceso se hace automáticamente, discutido por

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70



# 1. Terminar la instrucción actual

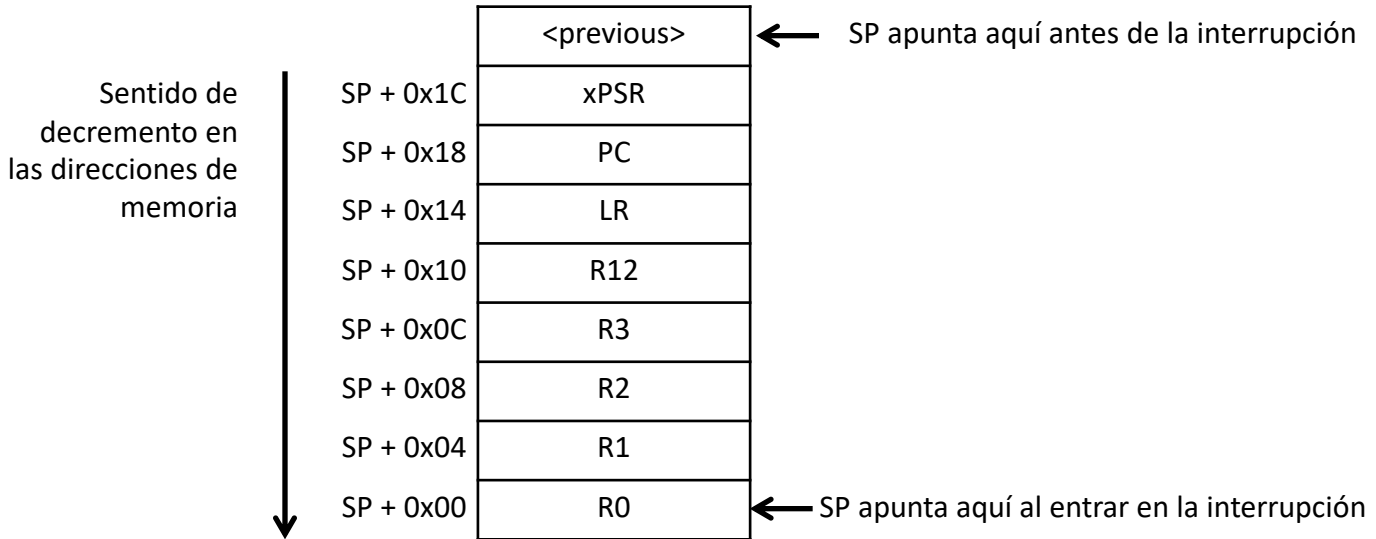
- La mayoría de las instrucciones son cortas y terminan rápidamente
- Sin embargo hay instrucciones que necesitan muchos ciclos de ejecución
  - Load Multiple (LDM), Store Multiple (STM), Push, Pop, MULS (32 ciclos en algunas CPUs basadas en la arquitectura)
- Estas instrucciones retrasarían la respuesta a la interrupción
- Si una de estas instrucciones se está ejecutando cuando entra una petición de interrupción, el procesador:
  - *abandona* la instrucción
  - responde a la interrupción
  - ejecuta la rutina de interrupción
  - retorna de la interrupción

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70



## [Entrando en una Interrupción] 2. Salvar el contexto en la pila



Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

[Entrando en una Interrupción]

### 3. Cambiar a modo privilegiado

Register	Value
<b>Core</b>	
R0	0x00000000
R1	0x00000000
R2	0x00000000
R3	0x0000008C
R4	0x40040000
R5	0x00000ECC
R6	0x2EE00000
R7	0x00000EAB
R8	0x9C7FB76F
R9	0x20000768
R10	0x00000ECC
R11	0x00000ECC
R12	0x0FFF00AA
R13 (SP)	0x20000FD8
R14 (LR)	
R15 (PC)	
<b>xPSR</b>	
<b>Banked</b>	
MSP	0x20000FD8
PSP	0xBEF5DFF0
<b>System</b>	
PRIMASK	0

El valor de SP se reduce porque los registros se salvaguardan en la pila

Saved R0

Saved R1

Saved R2

Saved R3

Saved R12

Address	sp
0x20000FD8:	00000000 00000000 00000000 0000008C 0FFF00AA
0x20000FEC:	00000A0B 00000A70 21000000 00000002 00000A0B
0x20001000:	621361E2 B510BD30 6810482F 04892101 492D4308
0x20001014:	20186008 FFD4F7FF 49292019 200562C8 F8EEF000

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

Cartagena99

## 4. Cargar PC con dirección de rutina de excepción

Memory Address	Value
0x0000_0000	Initial Stack Pointer
0x0000_0004	Reset
0x0000_0008	NMI_IRQHandler
...	
	IRQ0_Handler
	IRQ1_Handler
...	
Reset:	
...	
NMI_IRQHandler:	
...	
IRQ0_Handler:	

- El contador de programa se carga con la dirección de comienzo de la excepción correspondiente
- En memoria se muestra la tabla de vectores de excepción, en función de la interrupción concreta.

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70





# Tabla de excepciones

Exception number	IRQ number	Priority	Vector	Offset
			Initial SP	0x00
1		-3 (Highest)	Reset	0x04
2	-14	-2	NMI	0x08
3	-13	-1	HardFault	0x0C
4		Programable/ No disponible	Reserved	0x10
5				
6				
7				
8				
9				
10				
11	-5	Programable	SVCcall	0x2C
12		Programable/ No disponible	Reserved	
13		No disponible		
14	-2	Programable	PendSV	0x38
--				

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
 LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
 ---  
 ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
 CALL OR WHATSAPP: 689 45 44 70

## 4. Cargar PC con dirección de rutina de excepción (II)

Register	Value
<b>Core</b>	
R0	0x00000000
R1	0x00000000
R2	0x00000000
R3	0x0000008C
R4	0x40040000
R5	0x00000ECC
R6	0x2EE00000
R7	0x00000EAB
R8	0x9C7FB76F
R9	0x20000768
R10	0x00000ECC
R11	0x00000ECC
R12	0x0FFFFFFAA
R13 (SP)	0x20000FD8
R14 (LR)	0xFFFFFFFF9
R15 (PC)	0x00000ACC
xPSR	0x21000010
<b>Banked</b>	
<b>System</b>	
<b>Internal</b>	
Mode	Handler

```
42: void switch_isr(void) {  
0x00000ACC B510 PUSH {r4,lr}
```

PC se carga con la dirección inicial de la rutina de interrupción

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

Cartagena99

## 5. Cargar LR con el retorno de excepción

Register	Value
<b>Core</b>	
R0	0x00000000
R1	0x00000000
R2	0x00000000
R3	0x0000000C
R4	0x40040000
R5	0x0000ECC
R6	0x2EE00000
R7	0x0000EAB
R8	0x9C7FB76F
R9	0x20000768
R10	0x0000ECC
R11	0x0000ECC
R12	0x0FFFF0AA
R13 (SP)	0x20000FD8
R14 (LR)	0xFFFFFFFF9
R15 (PC)	0x0000ACC
xPSR	0x21000010
<b>Banked</b>	
MSP	0x20000FD8
PSP	0xBEF5DFF0
<b>System</b>	
PRIMASK	0

El valor 0xFFFFFFFF9 es un caso especial

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70



## 6. Cargar IPSR con el número de excepción

Register	Value
Core	
R0	0x00000000
R1	0x00000000
R2	0x00000000
R3	0x0000000C
R4	0x40040000
R5	0x0000ECC
R6	0x2EE00000
R7	0x0000EAB
R8	0x9C7FB76F
R9	0x20000768
R10	0x0000ECC
R11	0x0000ECC
R12	0x0FFFF0AA
R13 (SP)	0x20000FD8
R14 (LR)	0xFFFFFFFF9
R15 (PC)	0x0000ACC
xPSR	0x21000010
Banked	
MSP	0x20000FD8
PSP	0xBEF5DFF0
System	
PRIMASK	0

Exception number 0x10  
(interrupt number + 0x10)

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

## 7. Empezar la ejecución de la rutina

- La rutina de excepción se inicia, a no ser que se vea interrumpida por una excepción de mayor prioridad.
- La rutina puede salvaguardar registros adicionales en la pila
  - Por ejemplo, si la rutina puede llamar a una subrutina, LR y R4 deben salvaguardarse

```
42: void switch_isr(void) {  
0x00000ACC B510      PUSH      {r4,lr}
```

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# Salvaguardando los registros.

Register	Value
<b>Core</b>	
R0	0x00000000
R1	0x00000000
R2	0x00000000
R3	0x0000008C
R4	0x40040000
R5	0x00000ECC
R6	0x2EE00000
R7	0x00000EAB
R8	0x9C7FB76F
R9	0x20000768
R10	0x00000ECC
R11	0x00000ECC
R12	0x0FFFF0AA
R13 (SP)	0x20000FD0
R14 (LR)	0xFFFFFFFF
R15 (PC)	0x00000ACE
xPSR	0x21000010
<b>Banked</b>	
MSP	0x20000FD0
PSP	0xBEF50FF0
<b>System</b>	
PRIMASK	0
CONTROL	0x00

```
42: void switch_isr(void) {  
0x00000ACC B510 PUSH {r4,lr}
```

Address	sp
0x20000FD0:	40040000 FFFFFFF9 00000000 00000000 00000000
0x20000FE4:	0000008C 0FFFF0AA 0000A0B 0000A70 21000000
0x20000FF8:	00000002 0000A0B 00001F1 00001F1 00001F1
0x2000100C:	070001F1 070001F1 070001F1 070001F1 070001F1
0x20001020:	0C0001F1 0C0001F1 0C0001F1 0C0001F1 0C0001F1



Cartagena99

CLASES PARTICULARES, TUTORIAS TECNICAS ONLINE  
LLAMA O ENVIA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70



# INDICE ESPECÍFICO

- **Conceptos de Excepción e Interrupción**
  - Entrando en una Excepción
  - **Saliendo de una Excepción**
- **El núcleo de las Interrupciones**
- **Diseño de programa con Interrupciones**
  - Compartiendo datos de forma segura entre ISRs y otros hilos.

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70



# Saliendo de una Interrupción

## Secuencia del proceso de salida de una interrupción:

1. Ejecutar la instrucción que dispara el proceso de retorno de excepción.
2. Seleccionar la pila de retorno (MSP o PSP); restaurar el contexto (8 32-bit words) de la pila actual.
3. Reanudar la ejecución del código interrumpido cuando se lanzó la excepción.

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70





# 1. Inicio del proceso de retorno

- No existe una instrucción tal como “return from interrupt”.
- En su lugar se usa:
  - **BX LR** : Salto a la dirección almacenada en LR (carga el PC con el contenido de LR)
  - **POP ..., PC** – Restaura los valores desde la pila, entre ellos el PC
- ... con un valor especial **EXC\_RETURN**, que al cargarse en el PC dispara el proceso de retorno de la excepción
  - BX LR se usa si EXC\_RETURN está en LR
  - Si EXC\_RETURN se ha guardado en la pila, entonces se usa

FOR

Cartagena99

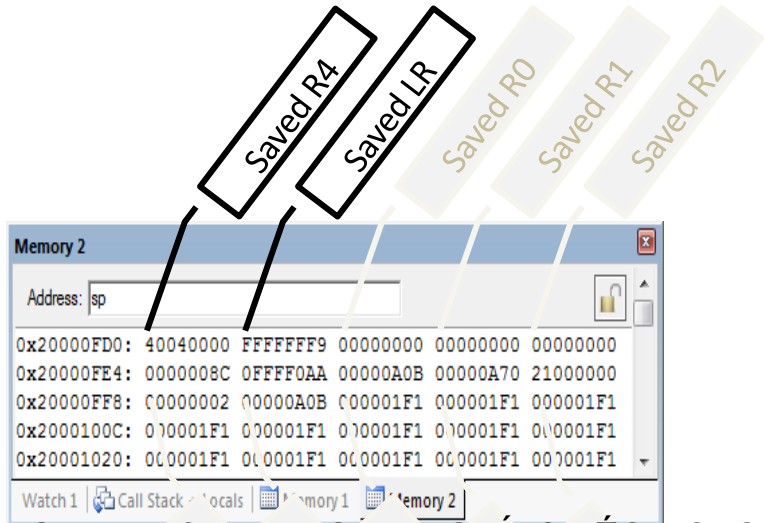
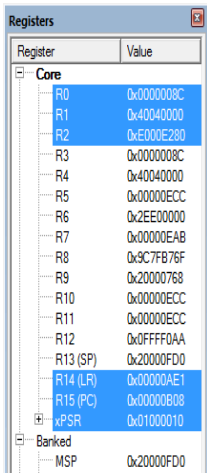
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# 1. Proceso de retorno. PC desde la pila

- R4: 0x4040\_0000
- PC: 0xFFFF\_FFF9

```
51: }  
->0x00000B08 BD10 POP {r4,pc}
```



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

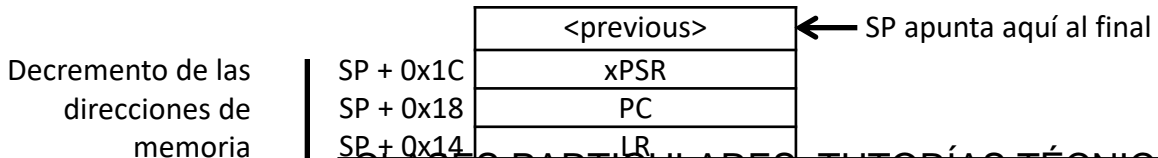
Cartagena99

## 2. Selección de la pila, restaurar contexto

- Dependiendo del valor de EXC\_RETURN se determina la pila de trabajo y el proceso de retorno

EXC_RETURN	Return Stack	Description
0xFFFF_FFF1	0 (MSP)	Return to exception handler with MSP
0xFFFF_FFF9	0 (MSP)	Return to thread with MSP
0xFFFF_FFFD	1 (PSP)	Return to thread with PSP

- Y se restauran los valores correctos de la pila seleccionada



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
 LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
 ---  
 ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
 CALL OR WHATSAPP:689 45 44 70

SP + 0x10 PC

[Saliendo de una Interrupción]

## 2. Siguiendo con el ejemplo...

- PC=0xFFFF\_FFF9, luego se utiliza la pila principal (MSP) y se retorna al hilo principal del programa de donde se salió.
- Se restauran automáticamente los valores almacenados en la pila a los registros.



Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

### 3. Reanudar la ejecución interrumpida

Register	Value
<b>Core</b>	
R0	0x00000000
R1	0x00000000
R2	0x00000000
R3	0x0000008C
R4	0x40040000
R5	0x00000ECC
R6	0x2EE00000
R7	0x00000EAB
R8	0x9C7FB76F
R9	0x20000768
R10	0x00000ECC
R11	0x00000ECC
R12	0x0FFFF0AA
R13 (SP)	0x20000FF8
R14 (LR)	0x00000A0B
R15 (PC)	0x00000A70
xPSR	0x21000000
<b>Banked</b>	
MSP	0x20000FF8
PSP	0xBEF50FF0
<b>System</b>	
PRIMASK	0
CONTROL	0x00

- Los registros que salvaguardó el procesador de forma automática al iniciarse la interrupción se restauran con su valor: R0, R1, R2, R3, R12, LR, PC, xPSR
- SP vuelve a tener el valor original
- Se vuelve al modo de trabajo hilo
- La siguiente instrucción a ejecutar es la situada en 0x0000\_0A70

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# INDICE ESPECÍFICO

- **Conceptos de Excepción e Interrupción**
  - Entrando en una Excepción
  - Saliendo de una Excepción
- **El núcleo de las Interrupciones**
- **Diseño de programa con Interrupciones**
  - Compartiendo datos de forma segura entre ISRs y otros hilos.

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# Tipos de interrupciones

- **Interrupciones Hardware**

- Asíncronas, no están relacionadas con el código que el procesador ejecuta en ese instante.
- Ejemplos: se recibe un carácter por el puerto serie, o se finaliza una conversión A/D.

- **Excepciones, fallos e interrupciones software**

- Síncronas, son el resultado de la ejecución de una instrucción
- Ejemplos: se produce desbordamiento en una operación, acceso a una posición de memoria inexistente.

- Podemos habilitar y deshabilitar la mayoría de las interrupciones (enmascarables), aunque algunas no (no enmascarables).

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# El Controlador de interrupciones (NVIC Nested Vectored Interrupt Controller)

- NVIC es el subsistema que maneja y prioriza las interrupciones externas
- Las interrupciones son un tipo de excepciones
  - Excepciones <16, Interrupciones: 16 a 16+N
- Modos
  - Modo hilo: entra con un Reset
  - Modo rutina: entra al ejecutar una excepción
- Se entra en un modo de ejecución privilegiado
- Punteros de pila
  - Main Stack Pointer, MSP
  - Process Stack Pointer, PSP
- Estados de excepción: Inactivo, Pendiente, Activo, Activo y pendiente

Port Module

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

Another Module





# Registros y estado del Controlador de interrupciones

- **Habilitado** – Permiten que se reconozcan las interrupciones
  - Se acceda a través de dos registros (bits a “1” para habilitar)
    - *Set enable con NVIC\_IUSER, clear enable con NVIC\_ICER*
  - Funciones con CMSIS:  
NVIC\_EnableIRQ(IRQnum),  
NVIC\_DisableIRQ(IRQnum)
- **Pendiente** – la interrupción ha sido solicitada pero aún no se ha permitido su comienzo
  - Funciones con CMSIS:  
NVIC\_SetPendingIRQ(IRQnum), NVIC\_ClearPendingIRQ(IRQnum)

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# Registros de máscara de interrupciones (Core Exception Mask Register)

- Similar al bit “Global interrupt disable” en otros micros.
- PRIMASK - Exception mask register (núcleo CPU)
  - Bit 0: PM Flag
    - “0” permite la activación de todas las excepciones
    - “1” evita la activación de todas las excepciones con prioridad configurable
  - Se modifica con las instrucciones CPS, MSR y MRS
  - Se usa para prevenir condiciones de carrera
- Funciones en CMSIS
  - void \_\_enable\_irq() – pone a cero el PM flag
  - void \_\_disable\_irq() – pone a uno el PM flag
  - uint32\_t \_\_get\_PRIMASK() – devuelve el valor de PRIMASK

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# Priorización

- Las excepciones son priorizadas con un número para poder arbitrar la respuesta ante peticiones de atención simultaneas (menor número = mayor prioridad)
- La prioridad de algunas excepciones es fija
  - Reset: -3, la mayor prioridad
  - NMI: -2
  - Hard Fault: -1
- La prioridad de otras excepciones (periféricos) es ajustable
  - Su valor se guarda en el “interrupt priority register” (IPR0-7)

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# Casos aplicados de priorización

- **Caso A:** Peticiones simultaneas de excepciones
  - Se atiende primero a la excepción con menor número de prioridad (interrupción más prioritaria)
- **Caso B:** Nueva petición de excepción mientras se está ejecutando la rutina de atención de otra
  - La prioridad de la nueva es **mayor** que la de la antigua
    - La nueva rutina se ejecuta y la antigua ha de esperar a que la nueva acabe
  - La prioridad de la nueva es **menor o igual** que la de la antigua
    - La nueva excepción queda en estado pendiente
    - La antigua rutina se ejecuta y acaba
    - Se restaura el anterior nivel de prioridad

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70



# Cómo priorizar las interrupciones

- El uso de las librerías **CMSIS** (Cortex Microcontroller Software Interface Standard) proporcionadas por ARM es la forma recomendada para programar los microcontroladores Cortex-M de forma que pueda portarse el programa entre dispositivos.
- CMSIS proporciona la función:  
**NVIC\_SetPriority(IRQn, priority)**  
para establecer la prioridad de las interrupciones.
- STM32 HAL proporciona llamadas para esta función

```
/* EXTI interrupt init*/
```

Cartagena99

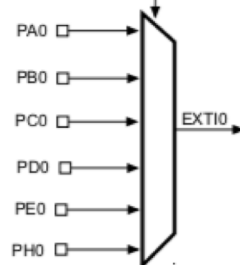
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

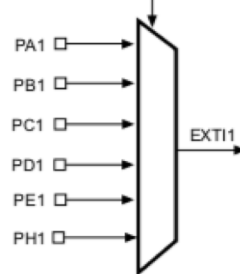
# Tipos de interrupciones externas (STM32F4)

- Podemos usar 16 líneas de interrupción externas (línea 0 a línea 15) para detectar eventos externos de los pines GPIO. Cada pin de cada puerto GPIO está conectado a una línea de interrupción externa con el mismo número.
- Ejemplo:
  - PA0, PB0, PC0, etc., se multiplexan a line0. Por lo tanto, solo puede configurar uno de estos pines para conectarse a la línea de interrupción 0.
  - PA0 y PA8 se multiplexan en diferentes líneas. Por lo que puede configurar estos pines para

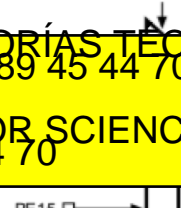
EXTI0[3:0] bits in the SYSCFG\_EXTICR1 register



EXTI1[3:0] bits in the SYSCFG\_EXTICR1 register



EXTI15[3:0] bits in the SYSCFG\_EXTICR4 register



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

## Tipos de interrupciones externas (STM32F4) (II)

- Para los manejadores (**handlers**) de interrupciones, STM32F4 tiene solo 7 manejadores de interrupciones para manejar las interrupciones externas de la línea 0 a la línea 15.
- Solo interrupciones externas de la line0 a line4 tienen su propio controlador IRQ.
- Line5 - line9 y line10 - line15 tienen el mismo controlador IRQ.

IRQ	IRQ Handler
EXTI0_IRQn	EXTI0_IRQHandler
EXTI1_IRQn	EXTI1_IRQHandler
EXTI2_IRQn	EXTI2_IRQHandler
EXTI3_IRQn	EXTI3_IRQHandler

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70



# INDICE ESPECÍFICO

- **Conceptos de Excepción e Interrupción**
  - Entrando en una Excepción
  - Saliendo de una Excepción
- **El núcleo de las Interrupciones**
- **Diseño de programa con Interrupciones**
  - Compartiendo datos de forma segura entre ISRs y otros hilos.

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70





# INDICE ESPECÍFICO

- **Conceptos de Excepción e Interrupción**
  - Entrando en una Excepción
  - Saliendo de una Excepción
- **El núcleo de las Interrupciones**
- **Diseño de programa con Interrupciones**
  - **Compartiendo datos de forma segura entre ISRs y otros hilos.**

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70



## Cuestiones a tener en cuenta

- ¿Cuanta carga de trabajo asignar a una rutina de interrupción? ¿Puede ser muy extensa?
- ¿Debería la rutina habilitar las interrupciones?
- ¿Cómo comunicar una rutina de interrupción con otros hilos del programa?
  - Datos compartidos (“data buffering”)
  - Integridad de los datos y condiciones de carrera (“race conditions”)

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# ¿Cuanta carga de trabajo asignar a una rutina de interrupción?

**Importante: la atención inmediata de las interrupciones retrasa la ejecución del resto del programa**

- Cuando tenemos varias rutinas de interrupción, o una rutina se ejecuta repetidamente, es conveniente realizar el trabajo crítico en la rutina de atención a la interrupción y almacenar los resultados parciales para un procesado posterior

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70



# Compartiendo datos de forma segura entre ISRs y otros hilos.

- Tipo “**Volatile data**” – se trata de un tipo de dato que puede ser actualizado en cualquier momento y sin control del programa que se ejecuta
- “**Non-atomic shared data**” – se trata de un tipo de dato que contiene múltiples campos. En ejecución se puede interrumpir su actualización, escritura o lectura, por lo que es vulnerable a las condiciones de carrera

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# Datos volátiles (“Volatile”)

- **Los compiladores asumen que las variables en la memoria no cambian espontáneamente y se optimizan en base a esa creencia**
  - No vuelven a cargar una variable de la memoria si la función actual no la ha cambiado
  - Leen variable de memoria en registro (acceso más rápido)
  - Escriben de nuevo en la memoria al final del procedimiento, o antes de una llamada a un procedimiento, o cuando el compilador se quede sin registros libres
- **Esta optimización puede fallar**
  - Ejemplo: leer un puerto de entrada, detectando una pulsación de una tecla
    - while (SW\_0); leerá de SW\_0 una vez y reutilizará ese valor
    - Generará un bucle infinito activado por SW\_0 siempre verdadero
  - Razones por las que puede fallar:

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

**cambia la variable**



# The Volatile Directive

- Hay que decirle al compilador qué variables pueden cambiar fuera de su control. Para ello:
  - Utilizar la palabra reservada “**volatile**” para forzar al compilador a recargar esta variable de la memoria en cada uso:  
**volatile unsigned int num\_ints;**

## ❖ Puntero a un volatile int

```
volatile int * var; // or  
int volatile * var;
```

- Ahora, cada lectura de una variable (por ejemplo, registro de estado) dará como resultado una instrucción LDR en lenguaje ensamblador

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# Non-Atomic Shared Data

- Queremos hacer un seguimiento de la fecha y hora
- Usamos interrupciones con temporizador de 1Hz
- Sistema
  - La estructura `current_time` rastrea el tiempo y los días desde algún evento de referencia
  - los campos de `current_time` se actualizan mediante la ISR y el

```
void GetDateTime(DateTimeType * DT){  
    DT->day = current_time.day;  
    DT->hour = current_time.hour;  
    DT->minute = current_time.minute;  
    DT->second = current_time.second;  
}
```

```
void DateTimeISR(void){  
    current_time.second++;  
    if (current_time.second > 59){  
        current_time.second = 0;  
        current_time.minute++;  
        if (current_time.minute > 59) {  
            current_time.minute = 0;  
            current_time.hour++;  
            if (current_time.hour > 23) {  
                current_time.hour = 0;  
                current_time.day++;  
            }  
        }  
        ... etc.  
    }
```

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70

# Ejemplo: comprobando la hora/fecha

- **Problema:**
  - Una interrupción en el momento equivocado provocará datos semi actualizados en DT
- Ejemplo de fallo:
  - `current_time` es {10, 23, 59, 59} (día 10, 23:59:59)
  - El código de la tarea llama a `GetDateTime ()`, que comienza a copiar los campos `current_time` en DT: día = 10, hora = 23
  - Se produce una interrupción del temporizador, que actualiza `current_time` a {11, 0, 0, 0}
  - `GetDateTime ()` reanuda la ejecución, copiando los campos restantes de `current_time` a DT: minuto = 0, segundo = 0
  - DT ahora tiene una marca de tiempo de {10, 23, 0, 0}.
  - ¡El sistema piensa que el tiempo acaba de saltar una hora!
- Problema fundamental: "**race condition**"
  - La prioridad permite a la ISR interrumpir otro código y posiblemente sobrescribir datos
  - **Se debe garantizar el acceso atómico (indivisible) al objeto.**

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70



# El problema en más detalle

- Se debe proteger cualquier objeto de datos que:
  - Requiere múltiples instrucciones para leer o escribir (acceso no atómico), y
  - Puede ser potencialmente escrito por una ISR
- ¿Cuántas tareas / ISR pueden escribir en el objeto de datos?
  - ¿Una? Entonces tenemos comunicación unidireccional
    - Debe asegurarse de que los datos no se sobrescriban a la mitad de la lectura.
      - El escritor y el lector no se interrumpen
  - ¿Más de una?
    - Debe asegurarse de que los datos no se sobrescriban a la mitad de la lectura.

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

– LOS ESCRITORES NO SE INTERRUMPEN

# Definiciones

- **Condición de carrera (Race Condition):** Comportamiento anómalo debido a una dependencia crítica inesperada de la temporización de los eventos. El resultado del código de ejemplo depende de la sincronización relativa de las operaciones de lectura y escritura.
- **Sección crítica (Critical section):** una sección de código que crea una posible condición de carrera. La sección de código solo puede ser ejecutada por un proceso a la vez. Se requiere algún mecanismo de sincronización en la entrada y salida de la sección crítica para garantizar el uso exclusivo.

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70

# Solución: Desactivar prioridades (preemption)

- Prevenir las prioridades dentro de la sección crítica
- Si un ISR puede escribir en el objeto de datos compartidos, se deben **deshabilitar las interrupciones**
  - ✓ guardar el estado de máscara de interrupción actual en m
  - ✓ desactivar interrupciones
- Restaurar el estado anterior posteriormente (es posible que las interrupciones ya se hayan deshabilitado por otro motivo)
- Utilice CMSIS-CORE para guardar, controlar y restaurar el estado de máscara de interrupción
- Evitar si es posible:
  - Deshabilitar interrupciones retrasa la respuesta a

```
void GetDateTime (DateTimeType * DT)
{
    uint32_t m;

    m = __get_PRIMASK();
    __disable_irq();

    DT->day = current_time.day;
    DT->hour = current_time.hour;
    DT->minute = current_time.minute;
    DT->second = current_time.second;
    __set_PRIMASK(m);
}
```

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70



# Resumen para datos compartidos

- En threads (hilos) y ISR, identifique los datos compartidos
- Determine qué datos compartidos son demasiado grandes para ser manejados atómicamente por defecto
  - Esto debe protegerse de la priorización (por ejemplo, deshabilitar las interrupciones, usar un mecanismo de sincronización RTOS)
- Declarar (e inicializar) las variables compartidas como volátiles en el archivo principal (o globals.c)
  - `volatile int my_shared_var=0;`
- Actualice extern.h para que estas variables estén disponibles para funciones en otros archivos.
  - `volatile int my_shared_var;`
  - `#include "extern.h"` en cada archivo que use las variables compartidas
- Cuando utilice datos compartidos largos (no atómicos), guarde, deshabilite y

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP:689 45 44 70



# Recursos recomendados

- Architecture Reference Manual:

<http://infocenter.arm.com/help/index.jsp>

- Cortex-M4 Technical Reference Manual:

[http://infocenter.arm.com/help/topic/com.arm.doc.ddi0439d/DDI0439D\\_cortex\\_m4\\_processor\\_r0p1\\_trm.pdf](http://infocenter.arm.com/help/topic/com.arm.doc.ddi0439d/DDI0439D_cortex_m4_processor_r0p1_trm.pdf)

- Cortex-M4 Devices Generic User Guide:

[http://infocenter.arm.com/help/topic/com.arm.doc.dui0553a/DUI0553A\\_cortex\\_m4\\_dgug.pdf](http://infocenter.arm.com/help/topic/com.arm.doc.dui0553a/DUI0553A_cortex_m4_dgug.pdf)

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE  
LLAMA O ENVÍA WHATSAPP: 689 45 44 70  
---  
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS  
CALL OR WHATSAPP: 689 45 44 70