

# **ARM: Primera Práctica**

## **Introducción a la arquitectura ARM y al lenguaje ensamblador**

Departamento de Ingeniería en Automática, Electrónica,  
Arquitectura y Redes de Computadores  
Universidad de Cádiz

José Alcántara Muñoz (Autor)  
Mercedes Rodríguez García (Supervisora)



En esta práctica utilizaremos el software **ARM Software Development Toolkit** de ARM Limited:

- Compuesto por el editor **ARM Project Manager** y el depurador **ARM Debugger**.
- Versión: 2.50
- Ejecutable en todas las versiones de 32 bits de Windows desde XP hasta Windows 7.

## 1. Ejercicios

A continuación contamos con un programa que simula una calculadora algo limitada (tenéis el código en *calculadora.s*), sobre la cual versarán los ejercicios.

En esta práctica utilizaremos los programas *ARM Project Manager* y *ARM Debugger*. Asimismo, se recomienda la lectura completa de las secciones 2.2, 3.1 y 3.2 y del capítulo 6 de la documentación antes de realizar la práctica.

1. Lee el código y contesta a las siguientes preguntas:
  - a) ¿Hay alguna subrutina en el mismo? Identifícalas indicando la etiqueta que les da nombre.
  - b) Explica para qué crees que sirve la instrucción MOV, que aparece varias veces en el código.
2. Necesitarás abrir la ventana de los registros en el ARM Debugger para poder continuar con la práctica.
3. Pon breakpoints en las líneas “MOV r1, r0”, “MOV r2, r0” y “SWI SWI\_Salir” para poder pausar el programa en el momento en que se tengan que introducir los valores.
4. Ejecuta el programa y prueba las distintas opciones (suma y multiplicación). ¿Qué sucede con el resultado de la multiplicación? Escribe la línea que falta para que el funcionamiento sea el correcto en el ARM Project Manager, compila y vuelve a ejecutar para comprobarlo.
5. Ejecuta el programa tras arreglar el problema, selecciona la multiplicación e introduce los operandos 0xFFFFFFFF y 0x0000000F, ¿el resultado es el esperado? Si no es así, ¿qué ha podido pasar?
6. SUMA y PROD son dos subrutinas del programa. ¿Qué instrucción se ha utilizado para poder ejecutar dichas subrutinas? Explica en qué difiere esta instrucción de la instrucción BL para realizar subrutinas y el porqué de esa diferencia.
7. ¿Podemos modificar el contenido del registro r14 en una subrutina? Explique por qué.

```

Calc,CODE,READWRITE
SWI_EscrCar EQU &0 ; código de impresión de carácter (0)
SWI_Salir EQU &11 ; código de impresión de salida del programa (11)
SWI_write0 EQU &2 ; código de impresión de cadena de caracteres (2)
SWI_ReadC EQU &4 ; código de lectura de un carácter (4)

ENTRY ; Punto de entrada del código
ADR r0, cad1 ; Obtenemos la dirección de la cadena1
SWI SWI_write0 ; Interrupción de Soft. para mostrar la cadena
; PARA INTRODUCIR EL DATO PONER PUNTO DE RUPTURA EN LA SIGUIENTE LINEA
; E INTRODUCIR EL DATO EN R0
MOV r1, r0 ; Movemos el valor introducido en r0 a r1

ADR r0, cad2 ; Obtenemos la dirección de la cadena2
SWI SWI_write0 ; Interrupción de Soft. para mostrar la cadena
; PARA INTRODUCIR EL DATO PONER PUNTO DE RUPTURA EN LA SIGUIENTE LINEA
; E INTRODUCIR EL DATO EN R0
MOV r2, r0 ; Movemos el valor introducido en r0 a r1

ADR r0, cad3 ; Obtenemos la dirección de la cadena3
SWI SWI_write0 ; Interrupción de Soft. para mostrar la cadena

SWI SWI_ReadC ; Interrupción de Soft. para leer un carácter de teclado
CMP r0, #49 ; ¿Se ha pulsado la tecla 1?
BLEQ SUMA ; ejecuta la rutina SUMA
CMP r0, #50 ; ¿Se ha pulsado la tecla 2?
BLEQ PROD ; ejecuta la rutina PROD

ADR r0, cad4 ; Obtenemos la direccion de la cadena3
SWI SWI_write0 ; Interrupción de Soft. para mostrar la cadena

; PARA INTRODUCIR EL DATO PONER PUNTO DE RUPTURA EN LA SIGUIENTE
; LÍNEA Y CONSULTAR EL RESULTADO EN R3
SWI SWI_Salir ; Sale del programa

SUMA ADD r3, r1, r2
MOV pc, r14

PROD
MUL r3, r1, r2

cad1 = "Intoduce operando 1 en registro r0 y pulsa F5", &0a, &0d, 0
cad2 = "Intoduce operando 2 en registro r0 y pulsa F5", &0a, &0d, 0
cad3 = "Elige operacion PULSANDO EN TECLADO el numero correspondiente:"
, &0a, &0d, "1. Sumar", &0a, &0d, "2. Multiplicar", &0a, &0d, 0
cad4 = &0a, &0d, "Programa terminado. Resultado en r3", &0a, &0d, 0
END

```

Figura 1: calculadora.s