

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



Índice

- 1 Introducción a ARM
 - Características de la arquitectura ARM
- 2 Ensamblador ARM
 - Instrucciones de procesamiento de datos
 - Instrucciones de control de flujo
- 3 Guión de prácticas

Introducción a ARM

La arquitectura ARM es una arquitectura que hoy en día está ganando terreno debido a su **eficiencia energética**. Se trata de una arquitectura pensada para tecnologías móviles (de ahí su eficiencia) y es utilizada actualmente en muchos **teléfonos móviles y tablets**. La familia ARM-Cortex es la más utilizada en la actualidad.

Figura : PlayStation Vita



Ejemplos



Figura : Google Nexus S

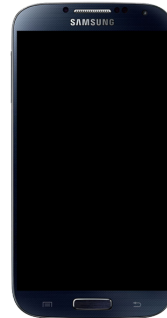


Figura : Samsung Galaxy S4

Índice

- 1 **Introducción a ARM**
 - Características de la arquitectura ARM
- 2 Ensamblador ARM
 - Instrucciones de procesamiento de datos
 - Instrucciones de control de flujo
- 3 Guión de prácticas

Características de la arquitectura ARM

La arquitectura ARM cuenta con las **características principales** de las llamadas **arquitecturas RISC**:

- **Una arquitectura de carga-almacenamiento:** Se cuenta con instrucciones específicas para los accesos a memoria, siendo este el único medio para acceder a memoria.
- **Instrucciones de longitud fija:** 32 bits.
- **Solo tres formatos de instrucciones.**

Además, esta arquitectura cuenta con otras características tales como la ejecución condicional de instrucciones o la capacidad de realizar desplazamientos a la vez que realizamos una operación. No obstante, estos conocimientos se escapan de los objetivos de esta práctica.

Características de la arquitectura ARM

La arquitectura ARM cuenta con las **características principales** de las llamadas **arquitecturas RISC**:

- **Una arquitectura de carga-almacenamiento:** Se cuenta con instrucciones específicas para los accesos a memoria, siendo este el único medio para acceder a memoria.
- **Instrucciones de longitud fija:** 32 bits.
- **Solo tres formatos de instrucciones.**

Además, esta arquitectura cuenta con otras características tales como la ejecución condicional de instrucciones o la capacidad de realizar desplazamientos a la vez que realizamos una operación. No obstante, estos conocimientos se escapan de los objetivos de esta práctica.

Características de la arquitectura ARM

La arquitectura ARM cuenta con las **características principales** de las llamadas **arquitecturas RISC**:

- **Una arquitectura de carga-almacenamiento:** Se cuenta con instrucciones específicas para los accesos a memoria, siendo este el único medio para acceder a memoria.
- **Instrucciones de longitud fija:** 32 bits.
- **Solo tres formatos de instrucciones.**

Además, esta arquitectura cuenta con otras características tales como la ejecución condicional de instrucciones o la capacidad de realizar desplazamientos a la vez que realizamos una operación. No obstante, estos conocimientos se escapan de los objetivos de esta práctica.

Registros del procesador

Al desarrollar programas a nivel de usuario, sólo se tienen en cuenta **15 registros** de propósito general de **32 bits** (del r0 al r14), el **contador de programa** (r15 o pc) y el **registro de estado** del programa actual (CPSR, *Current Program Status Register*).

El resto de registros sólo se usan para la programación a nivel del sistema y para el manejo de excepciones, por lo que no nos importan por ahora.

Ensamblador ARM

A continuación veremos de forma breve los distintos tipos de instrucciones que utilizaremos:

- **Instrucciones de procesamiento de datos:** Modifican el contenido de los registros mediante operaciones, comparaciones, etc.
- **Instrucciones de control de flujo:** Modifican el orden de ejecución de las instrucciones

Ensamblador ARM

A continuación veremos de forma breve los distintos tipos de instrucciones que utilizaremos:

- **Instrucciones de procesamiento de datos:** Modifican el contenido de los registros mediante operaciones, comparaciones, etc.
- **Instrucciones de control de flujo:** Modifican el orden de ejecución de las instrucciones

Índice

1 Introducción a ARM

- Características de la arquitectura ARM

2 Ensamblador ARM

- Instrucciones de procesamiento de datos
- Instrucciones de control de flujo

3 Guión de prácticas

Instrucciones de procesamiento de datos

Tenemos distintos tipos de instrucciones de procesamiento de datos:

- **Instrucciones aritmético-lógicas:** Suelen tener el formato `ADD r0, r1, r2`, donde el primer registro es el destino, el segundo es el primer operando, y el tercero es el segundo operando.
- **Instrucciones de movimiento de registros:** Siguen el formato `MOV r0, r1` y mueven el contenido del segundo registro al primero.
- **Operaciones de comparación:** éstas no producen un resultado, sino que sólo ponen los bits de código de condición (N, Z, C y V) en el CPSR de acuerdo a la operación en cuestión. Siguen el formato `CMP r1, r2`.

Instrucciones de procesamiento de datos

Tenemos distintos tipos de instrucciones de procesamiento de datos:

- **Instrucciones aritmético-lógicas:** Suelen tener el formato `ADD r0, r1, r2`, donde el primer registro es el destino, el segundo es el primer operando, y el tercero es el segundo operando.
- **Instrucciones de movimiento de registros:** Siguen el formato `MOV r0, r1` y mueven el contenido del segundo registro al primero.
- **Operaciones de comparación:** éstas no producen un resultado, sino que sólo ponen los bits de código de condición (N, Z, C y V) en el CPSR de acuerdo a la operación en cuestión. Siguen el formato `CMP r1, r2`.

Instrucciones de procesamiento de datos

Tenemos distintos tipos de instrucciones de procesamiento de datos:

- **Instrucciones aritmético-lógicas:** Suelen tener el formato `ADD r0, r1, r2`, donde el primer registro es el destino, el segundo es el primer operando, y el tercero es el segundo operando.
- **Instrucciones de movimiento de registros:** Siguen el formato `MOV r0, r1` y mueven el contenido del segundo registro al primero.
- **Operaciones de comparación:** éstas no producen un resultado, sino que sólo ponen los bits de código de condición (N, Z, C y V) en el CPSR de acuerdo a la operación en cuestión. Siguen el formato `CMP r1, r2`.

Multiplicación

Una forma especial de instrucciones de procesamiento de datos soporta la multiplicación:

```
MUL r4, r3, r2 ; r4 := (r3 x r2)
```

Este tipo de instrucción **no admite no admite operandos inmediatos y no permite que el registro fuente 1 sea a la vez el registro destino.**

Aunque se obtiene un resultado de 64 bits, sólo los 32 bits menos significativos se almacenan en el registro destino, perdiéndose el resto (aunque ARM también soporta el almacenamiento de la parte más significativa en un segundo registro).

Índice

1 Introducción a ARM

- Características de la arquitectura ARM

2 Ensamblador ARM

- Instrucciones de procesamiento de datos
- Instrucciones de control de flujo

3 Guión de prácticas

Instrucciones de control de flujo y subrutinas

Trataremos estas instrucciones con más detenimiento en la siguiente práctica.

Por el momento, sólo se necesita conocer la instrucción de control de flujo **BL**, que además de realizar un salto almacena el valor del contador de programa (pc o r15) en el registro de enlace (r14) para poder continuar luego el flujo normal de ejecución.

```
BL SUBRUT      ; salta a SUBRUT, r14 apunta a...  
                ; ... esta instrucción  
SUBRUT  ...  
MOV pc, r14    ; copiamos r14 en r15 para volver
```

Guión de prácticas

El resto de la práctica queda en vuestras manos. En la documentación contáis con información algo más amplia que probablemente tengáis que consultar.

Las herramientas a utilizar en esta práctica son **ARM Project Manager** y **ARM Debugger**, cuyas instrucciones de uso podéis encontrar también en la documentación.