Técnicas Digitales II

Arquitectura ARM Registros - Memoria

Arquitectura

Define el funcionamiento de un microprocesador y está compuesta por:

- El set de instrucciones.
- La ubicación de los operandos con respecto a la cpu y de que manera esta trabaja internamente.

Algunas arquitecturas ARM, x86, MIPS, SPARC, PowerPC

RISC vs CISC

- CISC (Complex Instruction Set Computer) o Computador con Conjunto de Instrucciones Complejas
 - Esta arquitectura tiene un set de instrucciones mas amplio.
 - Permite operaciones entre memoria o memoria/registro.
 - Motorola 68000, Zilog Z80 y toda la familia Intel x86, AMD x86-64.

RISC vs CISC

- RISC (Reduced Instruction Set Computer) o Computador con Conjunto de Instrucciones Reducidas
 - Set de instrucciones pequeño y de tamaño fijo, esto resulta en hardware para decodificar la instrucción mas simple, pequeño y rápido.
 - Solo existen instrucciones de carga y almacenamiento para acceder a memoria.
 - Suelen tener mayor cantidad de registros.
 - PowerPC, DEC Alpha, MIPS, ARM, SPARC

Set de Instrucciones

- Conjunto de instrucciones que realiza el microprocesador.
- Formada en general por operaciones básicas como sumas y restas.
- Cada instrucción indica la operación y los operandos a usar.
- Los operandos pueden ser registros, memorias o un valor inserto en la misma operación.
- Las instrucciones son codificadas como un número binario en un formato denominado código máquina.
- El ARM representa a cada instrucción con un número de 32 o 16 bits.

ARM

- Desarrollada por Acorn Computer Group en la década de 1980.
- Acrom junto con Apple crean en 1990 Advanced RISC Machine (ARM).
- La arquitectura ARM es licenciable. El negocio principal de ARM Holdings es la venta de núcleos.
- Es una de las arquitecturas mas difundidas (Wikipedia)
 - 98% de los mil millones de celulares vendidos anualmente.
 - 90% de microprocesadores RISC.

ARM

Empresas que poseen licencia ARM: (Wikipedia)

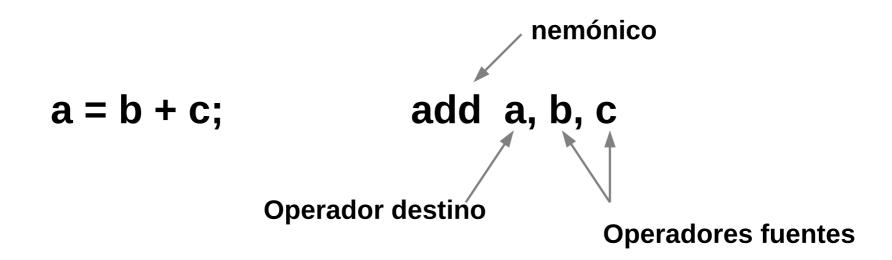
Alcatel-Lucent, Apple Inc., AppliedMicro, Atmel, Broadcom, Cirrus Logic, Digital Equipment Corporation, Ember, Energy Micro, Freescale, Intel (a través de DEC), LG, Marvell Technology Group, Microsemi, Microsoft, NEC, Nintendo, Nokia, Nuvoton, Nvidia, Sony, MediaTek, NXP (antes Philips Semiconductors), Oki, ON Semiconductor, Psion, Qualcomm, Samsung, Sharp, STMicroelectronics, Symbios Logic, Texas Instruments, VLSI Technology, Yamaha, y ZiiLABS.

Lenguaje Ensamblador

- Para facilitar la lectura de un humano las instrucciones del microprocesador, se representan en un formato simbólico denominado Lenguaje Ensamblador.
- Este lenguaje, representa el conjunto de instrucciones básicas de un microprocesador mediante nemónicos.
- A diferencias de lenguajes de mas alto nivel, cada microprocesador posee su propio lenguaje ensamblador.
- En general existe una relación de uno a uno desde las sentencias nemónicas al código máquina.

Lenguaje Ensamblador

- Cada instrucción especifica la operación a realizar y los operandos involucrados.
- Suma en lenguaje ensamblador



Lenguaje Ensamblador

Resta

$$a = b - c$$
;

Operación combinada

$$a = b + c - d;$$

add
$$t, b, c$$
; $t = b + c$

sub
$$a, t, d$$
; $a = t - d$

Registros

- Almacenan los operadores usados en las instrucciones.
- De rápido acceso pero de poca capacidad.
- Su tamaño define el tamaño de la arquitectura.
- ARM posee 16 registros.

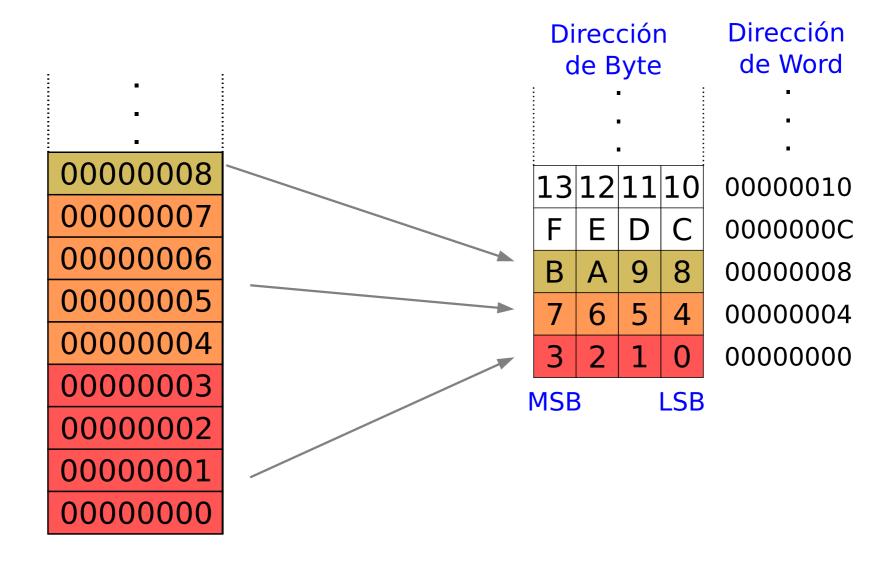
Registros

Registro	Uso
R0	Argumento / retorno de valor / variable temporal
R1 - R3	Argumento / variable temporal
R4 - R11	Variables Guardadas
R12	Variable Temporal
R13 (SP)	Puntero del Stack
R14 (LR)	Registro de Enlace o Link
R15 (PC)	Contador de Programa

Constantes / Inmediatos

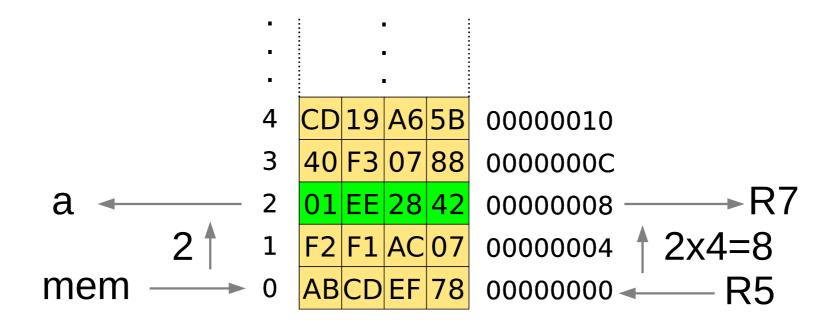
- Una operación puede ser realizada entre un registro y una constante.
- En ARM esta constante se llama valor inmediato por encontrarse en la misma instrucción y no requerir un nuevo acceso a memoria.

- En ARM todos las operaciones se realizan entre registros (o registro / inmediato)
- Para sortear la barrera de las 15 variables, se debe mover de la memoria a los registros y de los registros a la memoria.
- El ARM utiliza un modelo de memoria direccionada en bytes (byte-addressable).

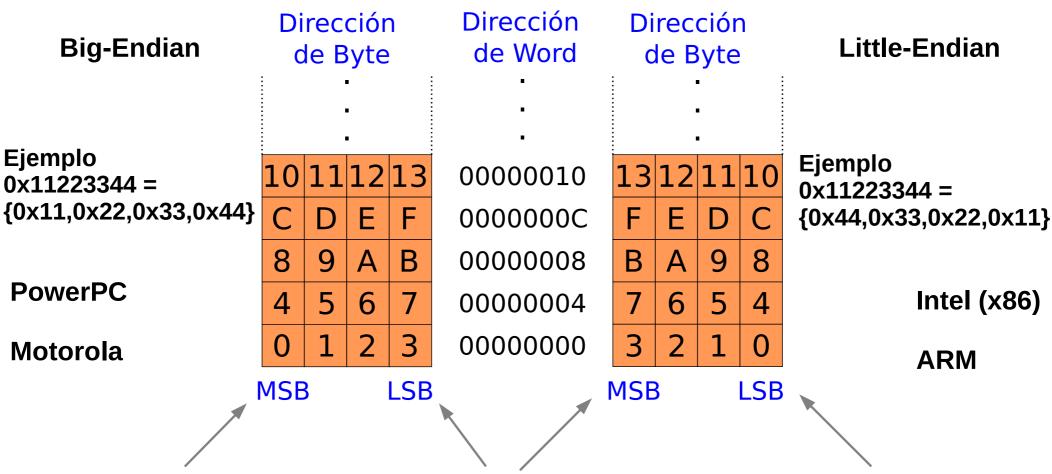


Dirección de Byte					Dirección de Word		Da	Número de Word		
	•				•		•			•
	•				•		•			•
	13	12	11	10	0000010	CD	19	A6	5B	4
	F	Е	D	С	000000C	40	F3	07	88	3
	В	Α	9	8	8000000	01	EE	28	42	2
	7	6	5	4	0000004	F2	F1	AC	07	1
	3	2	1	0	0000000	AB	CD	EF	78	0
	MSB LSB			LSB						

```
a = mem[2]; R7=a
MOV R5, #0
LDR R7,[R5,#8]
```



Endianness



Dirección de byte mas baja

Dirección de byte mas alta Dirección de byte mas baja Bibliografía

Harris & Harris. Digital design and computer architecture: ARM edition. Elsevier, 2015. Capítulo 6.

¿ Preguntas ?