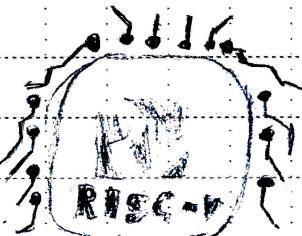


NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Saúl A. Sánchez A.	1 / 1	Micro	30/1/2024

# Title: Assembler en ARM y ESP32

Keyword	<b>Topic:</b> Lenguaje Assembler en ARM El Assembler es un lenguaje de programación de bajo nivel usado para programar directamente al procesador. Este lenguaje varía dependiendo del procesador. Por lo tanto, la arquitectura ARM utiliza un conjunto de instrucciones diferentes al resto, ya que la arquitectura RISC (Reduced Instruction Set Computer) de ARM, utilizada comúnmente en dispositivos móviles, sistemas integrados y algunos microcontroladores.
Questions	¿Qué es ARM? El lenguaje assembler en ARM surge de la necesidad de proporcionar a los programadores un medio para interactuar con esta estructura. ¿Qué es el lenguaje Assembler? La arquitectura ARM fue diseñada con el propósito de usarse en dispositivos pequeños y portátiles, donde la eficiencia energética y el rendimiento son fundamentales.
¿Qué es RISC?	 RISC

<b>Summary:</b> En resumen el lenguaje assembler es un lenguaje de bajo nivel y que varía dependiendo del procesador. En ARM se usa una versión diferente, diseñada para dispositivos más ligeros.
--

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Señor A. Sánchez A.	2/4	Micro	30/1/2024

Title: Assembler en ARM y ESP 32

Keyword	<p><b>Topic:</b> Sintaxis de Assembler en ARM.</p> <p>La sintaxis de las instrucciones ARM sigue un formato específico, con el uso de mnemotécnicos y operando. La sintaxis puede ser más limpia y uniforme en comparación con algunas arquitecturas.</p> <p>Algunos comandos verán:</p> <ul style="list-style-type: none"> <li>• LDR: Carga un valor en un Registro. Ej.: LDR R0, =Dx 1234 ; Carga el valor 0x1234 en R0</li> <li>• ;: El punto y coma se usa para hacer comentarios.</li> <li>• STR: Almacena un valor de un registro en la memoria. Ej.: STR R1, [R2] ; Almacenar R1 en R2</li> <li>• ADD: Para hacer suma.</li> <li>• SUB: Para restar.</li> <li>• MUL: Para multiplicación. Ej.: ADD R3, R4, R5 ; R3 = R4 + R5, lo mismo para el resto</li> <li>• AND, ORR, EOR: Para operaciones lógicas AND, OR y XOR Ej.: R11, R12, R13 ; R11 = R12 AND R13</li> </ul>
Questions	<p>¿Cómo hacer una suma en Assembler ARM?</p> <p>¿Qué es LDR en Assembler ARM?</p> <p>¿Qué otros comandos hay?</p>

Summary:	En resumen algunos de los comandos más importantes son: LDR, para cargar un valor de un registro, y STR, para guardar un valor en la memoria.
----------	---

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Saúl A. Sánchez A.	3/4	Micro	30/11/2024

## Title: Assembler en ARM y ESP32

<b>Keyword</b> ESP32 Xtensa	<b>Topic:</b> Lenguaje Assembler en ESP32
	<p>El lenguaje ensamblador para el ESP32 utiliza la arquitectura Xtensa. La arquitectura Xtensa tiene un conjunto de instrucciones similar al igual que ARM. El conjunto de instrucciones Xtensa es más complejo que el de ARM; debido a la falta de documentación detallada y oficial sobre el mismo.</p>
<b>Questions</b>  ¿Qué se necesita para programar en Assembler en ESP32?	<p>Para programar en assembler en el ESP32, lo común utilizan herramientas y software de desarrollo de Espressif Systems, el fabricante del ESP32.</p>
¿Qué es Xtensa?  ¿Qué es un ESP32?	<p>Código de ejemplo:</p> <pre> section .text     global start start:   ; iniciar el programa     addi a1,a0,10 ; a1 = a0+10     hlt ; terminar el programa   </pre>

**Summary:** En resumen, programar en Assembler en ESP32 es más complicado que en C/C++, y que muchas veces es perder el tiempo, por la poca optimización que obtendriamos.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Saúl A. Sánchez A.	4/4	Micro	30/1/2024

## Title: Assembler en ARM y ESP32

Keyword	Topic: Modos de direccionamiento a la memoria
Inmediata	Los modos de direccionamiento se refieren a la forma en que se especifica la dirección de memoria de los operandos en las instrucciones.
Registro	
Puntero Pila	
Modos de direccionamiento	Algunos son:
	<ul style="list-style-type: none"> <li>• Inmediato: La dirección del operando es un valor constante que se especifica en la instrucción.</li> <li>• Registro: La dirección del operando se guarda en un registro.</li> <li>• Base + Desplazamiento: La dirección del operando se forma sumando el contenido base de un registro con un desplazamiento constante.</li> <li>• Registro Indirecto: La dirección del operando se encuentra en un registro que contiene la dirección de memoria real.</li> <li>• Indicado de Registro: La dirección del operando se forma sumando el contenido de dos registros.</li> <li>• Puntero de Pila (Stack Pointer): La dirección del operando se toma desde la cima de la pila, que es administrada por el puntero de pila.</li> </ul>
Questions	<p>¿Qué son los modos de direccionamiento?</p> <p>¿Qué es el Puntero Pila?</p> <p>¿Cuales otros modos hay?</p>

**Summary:** Los modos de direccionamiento a la memoria, son las formas en donde guardamos la dirección de memoria, como lo son: Inmediata, registro, puntero de pila, etc.