

Instrucciones y Registros en Mips 32

Enyerber Flores

June 2, 2025

1 Introducción

El ensamblador es un lenguaje de bajo nivel que trabaja con registros y es necesario entender parte de la arquitectura para dominarlo. En el siguiente trabajo hablaremos de aspectos iniciales para empezar en este lenguaje, como lo son los **registros**, su funcionalidad y también las **etiquetas** que servirán para programar.

2 Registros

2.1 ¿Qué son?

Los registros son ubicaciones de almacenamiento ubicadas directamente en el cpu, y fueron diseñados para almacenar datos durante la ejecución de instrucciones.

2.2 ¿Cuáles son?

Son 32 registros:

- \$s0 - \$s7: Registros Guardados.
- \$t0 - \$t9: Registros Temporales.
- \$zero: Es siempre igual a 0.
- \$a0 - \$a3: Registros de argumentos para pasar parámetros.
- \$v0 - \$v1: Registros para retornar valores.
- \$gp: Puntero global.
- \$fp: Puntero de estructura.
- \$sp: Puntero de la pila.
- \$ra: Registro de la dirección de retorno.
- \$at: Maneja constantes grandes.
- \$k0 - \$k1: Manejo de excepciones.

3 Instrucciones en Mips 32

3.1 ¿Qué son?

Son los "ladrillos" que permiten construir Algoritmos controlando directamente el hardware. Existen muchos y cada uno tiene una funcionalidad particular, yo indagaré en los más importantes.

3.2 Lista de Instrucciones Básicas en Mips 32

- Aritmética:
 1. add: Guarda en un registro la suma de los valores en otros dos registros.
 2. sub: Guarda en un registro la suma de los valores en otros dos registros.
 3. mult: Guarda en un registro la multiplicación de los valores en otros dos registros.
 4. div: Guarda en un registro la división de los valores en otros dos registros.
- Transferencia de datos:
 1. lw: Carga una palabra de memoria a registro.
 2. sw: Carga una palabra de registro a memoria.
- Lógica:
 1. and: Guarda en un registro el resultado booleano de comparar con **y** dos registros.
 2. or: Guarda en un registro el resultado booleano de comparar con **o** dos registros.
 3. sll: Desplaza a la izquierda la cantidad que diga una constante.
 4. srl: Desplaza a la derecha la cantidad que diga una constante.
- Salto Condicional:
 1. beq: Comprueba la igualdad de dos registros y si se cumple, salta a otra dirección.
 2. bne: Comprueba si no hay igualdad en dos registros y si se cumple, salta a otra dirección.
- Salto Incondicional:
 1. j: Salta a una dirección señalada.
 2. jr: Retorna a un procedimiento.