

# Resumen

Samuel Reyna

June 3, 2025

## **Registro y Palabras de memoria**

Son unidades de almacenamiento. En MIPS, los registros almacenan datos para realizar operaciones aritméticas, mientras que las palabras de memoria almacenan estructuras de datos, tablas y registros desbordados. Otra característica a destacar es que los registros son más fáciles de utilizar, ya que no necesitas una instrucción específica de transferencia para acceder a los datos.

**Registros:** consta de 32 registros de 32 bits, s0–s7, t0–t9, zero, a0–a3, v0–v1, gp, fp, sp, ra, at.

**Palabras de memoria:** el campo que abarca un elemento se cuenta con números múltiplo de 4. Ejemplo: Memory[0], Memory[4], . . . , Memory[4294967292].

## **Instrucciones**

MIPS contiene distintos tipos de instrucciones, aunque las principales las podemos categorizar de la siguiente manera:

- **aritméticas:** son operaciones de suma (add), resta (sub), división (div) y multiplicación (mult), en donde se operan solo dos datos para guardar el resultado en un tercero. de aquí se le añade una especie de sufijo, esto como muestra que esa operación realiza una acción especial, como sería la "operación" inmediata (i), "operación" sin signo (u), "operación" FP simple (.s) o FP sobre (.d).
- **Transferencia de datos** son instrucciones que permiten mover información entre registros y memoria. Se dividen en dos tipos de instruc-

ciones, cargar (load) y almacenar (store), pueden ser palabra (lw/sw), media palabra (lh/hs), un byte (lb/sb), entre otros.

- **Lógicas:** realizan operaciones bit a bit sobre registros, los cuales nos permiten ejecutar operaciones booleanas (and/or/nor), y operaciones de desplazamiento (sll/srl).
- **salto condicional:** permiten cambiar el flujo del programa solo si se cumple una condición específica. Estos se pueden dividir en dos tipos de instrucciones, saltos (branch) que comparan si dos datos son iguales (beq) o no (bne), y comparaciones (set) los cuales mediante los saltos se define si un dato es menor que el otro dato (slt) o constante (slti).
- **salto incondicional:** similar al punto anterior, pueden cambiar el flujo del programa pero estas instrucciones no necesitan evaluar ninguna condición. en este caso el salto (jump) no necesita de alguna condición basta con apuntar a la dirección de destino, como sería retornar de un procedimiento (jr) o enlazarlo (jal).