Resumen: Evolución de la Computación y Arquitectura MIPS 32

Mike Gonzalez - 30281540

May 30, 2025

1 Introducción a la Computación

Desde sus inicios, la computación ha sido uno de los pilares fundamentales en el desarrollo tecnológico y social. La industria de las tecnologías de la información ha evolucionado rápidamente, impactando desde el ámbito científico hasta el cotidiano.

1.1 Innovación y Avance Tecnológico

La industria de los computadores se ha caracterizado por una constante innovación acelerada. Cada generación de hardware ha llevado a mejoras significativas en rendimiento y reducción de costos, permitiendo la aparición de nuevas aplicaciones. A lo largo de las décadas, la computación ha influenciado diversas áreas:

- **Automóviles**: Implementación de microprocesadores para gestión de motores y sistemas de seguridad.
- **Telefonía móvil**: Expansión de la conectividad global gracias al desarrollo de sistemas embebidos y redes eficientes.
- **Genómica**: Reducción de costos para el análisis de ADN mediante la computación avanzada.
- **Internet**: Surgimiento de herramientas como la World Wide Web y motores de búsqueda, revolucionando el acceso a la información.

1.2 Tipos de Computadores

Los computadores han evolucionado en diferentes formas y propósitos:

- **Computadores de sobremesa**: Uso personal con software generalizado.
- **Servidores**: Sistemas diseñados para manejar cargas pesadas en redes y procesamiento de datos.

- **Supercomputadores**: Computadores de alto rendimiento utilizados en cálculos científicos avanzados.
- **Computadores embebidos**: Integrados en dispositivos electrónicos para tareas específicas, como teléfonos y automóviles.

2 Arquitectura MIPS 32

MIPS 32 es una arquitectura basada en el diseño RISC (Reduced Instruction Set Computing), optimizando la eficiencia del procesamiento mediante un conjunto reducido de instrucciones.

2.1 Registros en MIPS 32

Los registros juegan un papel fundamental en la manipulación de datos dentro de la arquitectura. Algunos de los principales registros incluyen:

- `$zero`: Registro constante con valor 0.
- `$at`: Utilizado por el ensamblador.
- `$t0 - $t9`: Registros temporales para cálculos intermedios.
- `$s0 - $s7`: Valores que se deben conservar entre llamadas de funciones.
- `$gp, $fp, $sp, $ra`: Registros de propósito específico para gestión de memoria y flujo de ejecución.

2.2 Organización de Memoria en MIPS 32

La memoria en MIPS se organiza en direcciones de bytes, con estructuras alineadas en palabras de 4 bytes. Las instrucciones de carga y almacenamiento permiten el acceso eficiente a los datos.

2.3 Conjunto de Instrucciones en MIPS 32

Las instrucciones en MIPS 32 se dividen en varias categorías:

2.3.1 Instrucciones Aritméticas

- **Suma**: ‘add $s1, $s2, $s3‘ → Calcula la suma de ‘$s2‘ y ‘$s3‘, almacenando el resultado en ‘$s1‘.
- **Resta**: ‘sub $s1, $s2, $s3‘ → Resta ‘$s2‘ y ‘$s3‘, guardando el resultado en ‘$s1‘.
- **Suma inmediata**: ‘addi $s1, $s2, 100‘ → Suma un valor constante a ‘$s2‘.

2.3.2 Instrucciones de Transferencia de Datos

- **Cargar palabra**: ‘lw $s1, 100($s2)‘ → Obtiene un valor de memoria y lo guarda en ‘$s1‘.
- **Guardar palabra**: ‘sw $s1, 100($s2)‘ → Almacena el valor de ‘$s1‘ en memoria.

2.3.3 Instrucciones Lógicas

- **AND bit-a-bit**: ‘and $s1, $s2, $s3‘ → Operación lógica entre ‘$s2‘ y ‘$s3‘.
- **OR bit-a-bit**: ‘or $s1, $s2, $s3‘ → Aplica OR lógico entre ‘$s2‘ y ‘$s3‘.

2.3.4 Control de Flujo

- **Saltos condicionales**: ‘beq $s1, $s2, etiqueta‘ → Salto si ‘$s1 == $s2‘.
- **Saltos incondicionales**: ‘j etiqueta‘ → Salta a la dirección indicada.

3 Ejemplo de Código en MIPS

```
  .data
mensaje: .asciiz "Hola Mundo"

.text
.globl main
main:
    # Imprimir cadena
    li $v0, 4          # C digo de syscall para imprimir una cadena
    la $a0, mensaje    # Cargar direccion del mensaje
    syscall            # Llamar a la syscall

    # Finalizar el programa
    li $v0, 10         # C digo de syscall para terminar la ejecucion
    syscall
```