

Práctica No. 3, 4

Diseño e implementación de un procesador monociclo Programación MIPS

1. Objetivo

- Analizar los requerimientos de una arquitectura de conjunto de instrucciones (ISA) MIPS que permita la implementación de un procesador monociclo que ejecute la ISA.
- Codificar, ensamblar y simular un programa para verificar el comportamiento correcto del procesador.
- Emplear herramientas de software para el diseño y la simulación de computadores digitales.

2. Descripción

Se implementa el datapath del MIPS de 32 BITS que permita ejecutar las instrucciones de 32 bits que se muestran en la Tabla 1, si se desea agregar instrucciones adicionales lo puede hacer. La figura 1 muestra un datapath básico, si se necesita añadir otros componentes adicionales lo puede hacer. La arquitectura tendrá la capacidad para ejecutar algoritmos funcionales.

Tabla 1. Lista de instrucciones

Tipo de instrucción	Nemónico	Formato	Ejemplo	Operación
Acceso a memoria	lw	I	lw rd, offset(rs)	$R[rd] = M[R[rs] + \text{signExtImm}]$
	sw	I	sw rd, offset(rs)	$M[R[rs] + \text{signExtImm}] = R[rd]$
Aritmético-lógicas	add	R	add rd, rs, rt	$R[rd] = R[rs] + R[rt]$
	sub	R	sub rd, rs, rt	$R[rd] = R[rs] - R[rt]$
	and	R	and rd, rs, rt	$R[rd] = R[rs] \& R[rt]$
	slt	R	slt rd, rs, rt	$R[rd] = (R[rs] < R[rt]) ? 1 : 0$
	addi	I	addi rd, rs, INM	$R[rd] = R[rs] + \text{INM}$
Salto condicional	beq	I	beq rs, rd, label	If($R[rs] == R[rt]$) $PC = PC + 4 + \text{BranchAddr}$
Salto y enlace (OPCIONAL)	jal	J	jal funcion	$ra = PC + 4$ $PC = (PC \& 0xf0000000) (\text{target} << 2)$
Salto a registro (OPCIONAL)	jr	R	jr rs	$PC = R[rs]$

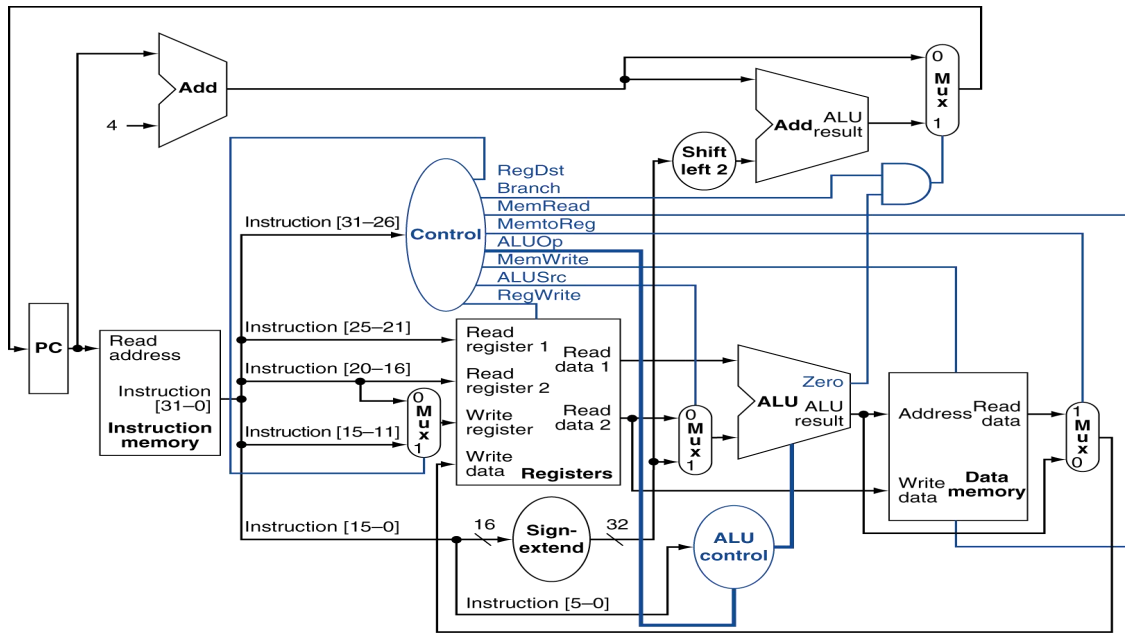


Figura 1. Datapath base y Unidad de Control del MIPS 32 bits

Tabla 2. Lista de modificaciones

La modificación que corresponde implementar se halla realizando la operación Módulo 8 al número de su equipo.

Modificación	Descripción
0	Los opcodes de LW y de la SW serán los 2 últimos dígitos del documento de cada uno de los integrantes respectivamente (%64).
1	Los opcodes de ADDI y de BEQ serán los 2 últimos dígitos del documento de cada uno de los integrantes respectivamente (%64).
2	Los opcodes de LW y de ADDI serán los 2 últimos dígitos del documento de cada uno de los integrantes respectivamente (%64).
3	Los opcodes de LW y de BEQ serán los 2 últimos dígitos del documento de cada uno de los integrantes respectivamente (%64).
4	Los opcodes de SW y de BEQ serán los 2 últimos dígitos del documento de cada uno de los integrantes respectivamente (%64).
5	Los opcodes de SW y de ADDI serán los 2 últimos dígitos del documento de cada uno de los integrantes respectivamente (%64).
6	los valores de function para ADD y SUB serán los 2 últimos dígitos del documento de cada uno de los integrantes respectivamente (%64).
7	los valores de function para ADD y AND serán los 2 últimos dígitos del documento de cada uno de los integrantes respectivamente (%64).

3. Procedimiento

PARTE 1:

Diseño e implementación del procesador usando Logisim y Prueba funcional

- a) Cada equipo implementará una versión modificada del procesador MIPS visto en clase. La modificación consiste en cambiarle los valores del opcodes o del campo function de algunas instrucciones tal como se establece en la Tabla 2.
- b) Implemente un procesador monociclo de la arquitectura de 32 bits, incluyendo la ruta de datos y las señales de control, para poder ejecutar **cada una** de las instrucciones de la tabla 1.
- c) Diseñe una unidad de control para el procesador monociclo que permita ejecutar las instrucciones de la tabla 1 teniendo en cuenta la asignación de la tabla 2, e implemente la CPU en su totalidad.
- d) Pruebe una a una las instrucciones de la tabla 1 en la arquitectura implementada.

PARTE 2:

Problema a resolver

- e) En el apendice A se encuentra el problema asignado a cada equipo. Deben realizar el pseudocódigo del algoritmo que resuelve el problema, luego debe ser codificado en ensamblador, traducido a lenguaje de máquina y ejecutado en el procesador.
- f) Puede usar el MARS como herramienta de apoyo pero tenga en cuenta que el código máquina producido por el MARS no es, necesariamente, 100% compatible con su arquitectura. Realice los ajustes necesarios al código máquina.
- g) **OPCIONAL:** El código tiene el llamado por lo menos a un método.

PARTE 3:

- h) Realice un análisis teórico de todos los accesos a memoria en la ejecución de su programa, y halle la cantidad de misses y de hits suponiendo que se tiene una cache de 4 bloques y que el bloque es de una palabra.

4. Requerimientos adicionales

Para el desarrollo de la práctica se imponen las siguientes condiciones:

- Ser riguroso al momento de dar solución a los apartados del procedimiento. Todas las decisiones de diseño deben estar ampliamente justificadas.

- Los demás componentes necesarios para implementar el procesador y que se considere no están disponibles en las bibliotecas de componentes de **Logisim**, deben ser diseñados con la misma herramienta.
- El código ensamblador debe estar comentado. Se valorará la creatividad y elegancia en la programación a este nivel.

5. Informe

Cada grupo de trabajo debe realizar un informe escrito que incluya:

- a. Una descripción detallada de todo el proceso de diseño y simulación del procesador, incluyendo la justificación de las decisiones de diseño tomadas durante el desarrollo de la práctica.
- b. Una descripción detallada de todo el proceso de diseño y simulación del algoritmo, incluyendo la justificación de las decisiones de diseño tomadas durante el desarrollo de la práctica.
- c. Los resultados de ejecución.
- d. Las observaciones y **conclusiones** pertinentes del trabajo.

Este documento debe estar en formato **PDF** y ser subido a la plataforma, acompañado de los archivos de diseño de **Logisim Evolution**, antes del cierre del plazo de entrega. El informe tiene un peso del 30% en la calificación global de la práctica.

6. Sustentación

Ambos miembros del equipo de trabajo deben demostrar un dominio completo del desarrollo de la práctica, mediante un video explicativo con una duración máxima de diez minutos, en el que intervengan de manera equitativa.

La entrega del video se debe realizar como un enlace de acceso al mismo incluido en el informe, y no como un archivo adjunto. El profesor podrá citar a algunos equipos de trabajo para que respondan preguntas que permitan aclarar o ampliar la información suministrada en el video con la sustentación. La sustentación tiene un peso del **70%** en la calificación global de la práctica.

Debe quedar claro los conocimientos adquiridos, cubriendo los siguientes aspectos:

- Implementación del procesador (Datapath + Control).
- Programa de simulación y de prueba del procesador.
- Codificación del programa en ensamblador.
- Manejo de la herramienta de diseño y simulación.

NOTA: No cumplir con alguno de los requerimiento solicitados (Informe formato PDF, adjuntar archivos del Logisim, adjuntar enlace del video dentro del informe) puede generar la no calificación de la práctica.

7. Bibliografía

- Notas de Clase
- Computer organization and design. The hardware/software interface, 5th ed.

APENDICE A

El número del problema que le corresponde a su equipo es el resultado de calcular **[(número de equipo) MOD 15] + 1**

NOTA: El vector debe tener un marcador de finalización, el programa no puede saber previamente el tamaño del vector.

OPCIONAL: Se valorará adicional si el programa contiene la implementación y el llamado por lo menos a una función.

Lista de problemas:

En un vector, almacenado en la memoria, halle:

1. La operación $10x+3$, x es el último elemento del vector.
2. La operación $20x$, x es el penúltimo elemento del vector.
3. La operación $x\%2$, x es el primer elemento del vector.
4. La operación $30x$, x es el último elemento del vector.
5. La operación $5x$, x es el elemento de la mitad del vector.
6. Multiplique el primer elemento con el último elemento.
7. En cuál posición está el elemento menor del vector.
8. Cuántos elementos son iguales al primer elemento.
9. Cuántos elementos son pares.
10. Multiplique los dos últimos elementos.
11. Cuántos elementos son impares.
12. Hallar la suma de los elementos en las posiciones impares del vector.
13. En cuál posición está el elemento mayor.
14. Cuántos elementos son iguales al segundo elemento.
15. Sume los elementos del medio del vector (2 o 3 elementos).