



Tecnológico de Monterrey

Instituto Tecnológico y de Estudios Superiores de Monterrey
(Campus Guadalajara)

Diseño de sistemas en chip (Gpo 501)
Implementación de un procesador RISC-V single cycle

Alumna:

Elizabeth Jáuregui Zárate (A01253381)

Fecha de entrega:
25/05/2025

Estructura del proyecto

Módulos:

1. main.v:
 - a. Entradas:
 - i. clk
 - b. Salidas (unicamente para poder probar el funcionamiento correcto del sistema, no son requeridas)
 - i. [31:0] debug_pc, debug_instruction
 - c. Funcionamiento:
 - i. Módulo top, donde se instancian todos los módulos que componen el Single Cycle de un procesador RISC-V. En este módulo se unen mediante señales auxiliares y registros. De igual manera, en este módulo se simula el funcionamiento de los multiplexores y el program counter.
2. instruction_memory.v:
 - a. Entradas:
 - i. [31:0] A
 - b. Salidas
 - i. [31:0] RD
 - c. Funcionamiento:
 - i. Simula una memoria de instrucciones de solo lectura (ROM) con 32 palabras de 32 bits. Lee instrucciones a partir de un archivo “instructions.mem”.
3. register_file.v:
 - a. Entradas:
 - i. clk, WE3
 - ii. [4:0] A1, A2, A3
 - iii. [31:0] WD3
 - b. Salidas
 - i. [31:0] RD1, RD2
 - c. Funcionamiento:
 - i. Implementa un banco de registros con dos puertos de lectura y uno de escritura. Tiene 32 registros de 32 bits, como en el ISA RISC-V.
4. extend.v:
 - a. Entradas:
 - i. [31:0] entrada
 - ii. [1:0] immSrc
 - b. Salidas

- i. [31:0] immExt
- c. Funcionamiento:
 - i. Extiende inmediatos de 12 a 32 bits con signo según el tipo de instrucción (I, S, B, J).

5. ALU.v:

- a. Entradas:
 - i. [31:0] SrcA, SrcB
 - ii. [2:0] alu_control
- b. Salidas
 - i. [31:0] alu_result
 - ii. zero
- c. Funcionamiento:
 - i. Realiza operaciones aritméticas y lógicas entre dos fuentes de 32 bits, basadas en una señal de control.

6. data_memory.v:

- a. Entradas:
 - i. clk, WE
 - ii. [31:0] A, WD
- b. Salidas
 - i. [31:0] RD
- c. Funcionamiento:
 - i. Implementa una memoria de datos que permite lecturas y escrituras sincronizadas con el flanco de subida del reloj. La escritura ocurre cuando WE (write enable) está activo.

7. control_unit.v:

- a. Entradas:
 - i. clk, funct7_5, zero
 - ii. [6:0] op
 - iii. [2:0] funct3
- b. Salidas
 - i. PCSrc, resultSrc, memWrite, regWrite, ALUSrc
 - ii. [2:0] alu_control
 - iii. [1:0] immSrc
- c. Funcionamiento:
 - i. Genera las señales de control necesarias para ejecutar instrucciones RISC-V. Se compone de dos submódulos, uno que decodifica el opcode y genera señales de alto nivel y otro que decodifica detalles de la ALU a partir de funct3 y funct7.