

Manuel Arambula

top

Función: Es el módulo principal que instancia todos los demás.

Conexiones: clk, reset, y señales intermedias entre los módulos.

Qué revisar: Asegúrate de que todos los módulos estén conectados correctamente desde aquí.

2. instruction_memory

Función: Contiene el programa en memoria. Devuelve la instrucción según la dirección.

Entradas: addr (dirección de instrucción).

Salidas: instruction.

Importante: Se inicializa con \$readmemh("...").

3. control_unit

Función: Decodifica la instrucción y genera señales de control:

reg_write, alu_src, mem_read, mem_write, mem_to_reg, branch, etc.

Entrada: opcode (instruction[6:0]).

Salida: señales de control.

4. register_file

Función: Tiene 32 registros (x0 a x31). Lee dos y escribe uno.

Entradas:

rs1, rs2 → índices de lectura.

rd → índice de escritura.

write_data → dato a escribir.

reg_write → habilita la escritura.

Salidas:

read_data1, read_data2.

5. immediate_generator

Función: Genera el valor inmediato según el tipo de instrucción (I, S, B, etc.).

Entrada: instrucción completa.

Salida: immediate.

6. alu_control

Función: Según el funct3, funct7 y el tipo de instrucción, decide la operación que hará la ALU.

Entrada: alu_op, funct3, funct7.

Salida: código de operación para la ALU (alu_control).

7. alu

Función: Ejecuta operaciones aritmético-lógicas.

Entradas:

input_a (registro rs1),

input_b (registro rs2 o inmediato),

alu_control (tipo de operación).

Salida:

alu_result y zero (para branch).

8. data_memory

Función: Simula la RAM para instrucciones tipo lw, sw.

Entradas:

addr, write_data, mem_write, mem_read.

Salida: read_data.

9. muxes (multiplexores)

Función: Seleccionan:

entre rs2 o immediate (para ALU input B),

entre alu_result o mem_read_data (para escritura en registro),

entre PC+4 o PC + immediate (para saltos).

Controlados por: alu_src, mem_to_reg, branch.

10. pc_register

Función: Guarda el Program Counter (PC).

Actualiza con: pc_next cada ciclo de reloj.

Se usa para: obtener instruction_memory[PC].

11. adder

Función: Suma el PC + 4 o PC + immediate según si hay salto.

Importante para: beq, jal, jalr.

Test bench hace lo esperado

