Advanced Computer Architecture

Computer Assignment 2: MIPS Pipeline – 2 way Superscalar

Amirmahdi Joudi: 810101325 **Negin Safari: 810101339**

Abstract:

In this assignment, a 2-way superscalar pipeline design of MIPS processor with support a subset of instructions is designed. This processor has 5 stages: Instruction Fetch (IF), Instruction Decode (ID), Execution (EX), Memory Access (MEM), and Write Back (WB). There are 2 ways: One supports integer and branch instructions; other way supports integer and memory instructions. It also supports forwarding and hazard unit.

Description:

The datapath for this design is shown in Figure 1. Instructions and Data memories are considered independent, so there is not memory access conflict. To prevent pipeline from stalling because of branch instructions, a comparator for branch condition check is put in ID stage to calculate equality earlier, so hazard unit stalls pipeline just for memory read dependency and parallel lanes hazards. Instructions supported by this design are shown in Table 1. The controller of this design is a simple lookup table that searches ‘opcode’ and ‘func’ bits to issue specific controlling signals. This architecture has the following specifications:

- When 2 instructions enter the processor at the same time while they have data dependency, the second instruction faces stall for one cycle.
- When two memory instructions or two branch instructions enter at the same, the second instruction faces stall for one cycle.
- There are six points for data forwarding in 2 ways. Data forwarding from one way to another is available.
- If there is a memory instruction and an integer instruction without data dependency but in the wrong way, instructions are swapped.
- If there is a branch instruction and an integer instruction without data dependency but in the wrong way, branch instruction faces stall for one cycle.
- On ordered branch taken, both instructions are flushed.
- If both instructions write in same registers, second instruction faces stall.
- Instructions are committed in order.

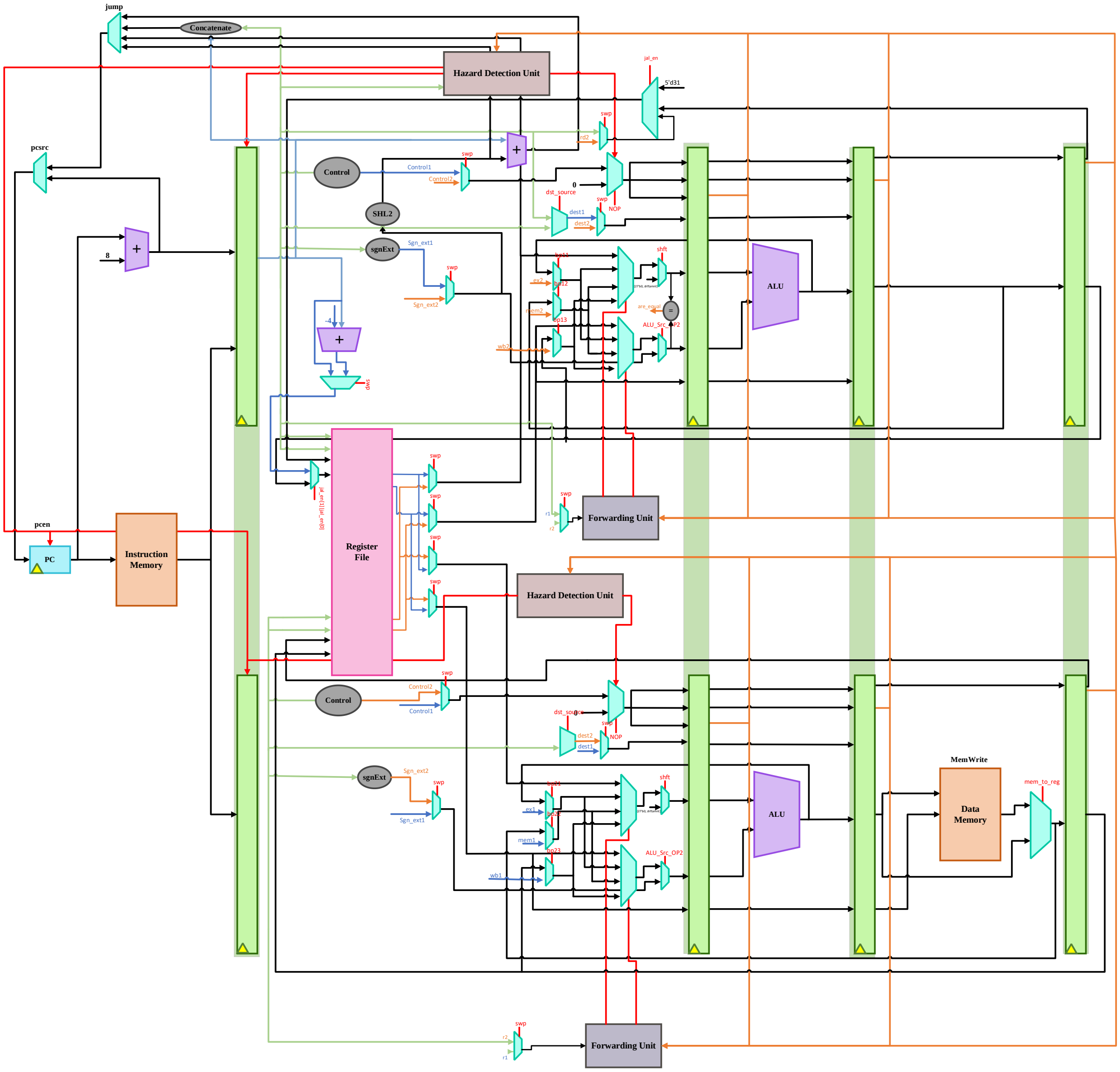


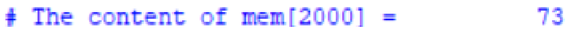
Figure 1. MIPS 2-way superscalar datapath

Table 1. Bolded instructions are implemented in this design

R-Type	**add, addu, sub, subu, slt, sltu, and, or, xor, nor, sll, srl, sla, sllv, arlv, slav, jr, jalr,** mult, multu, div, divu, mfhi, mthi, mflo, mtlo
I-Type	**addi, addiu, slti, sltiu, andi, ori, xori, lui, lw, sw, beq, bne**
J-Type	**j, jal**
FR-Type	add.s, sub.s, mul.s, div.s, abs.s, neg.s, c.eq.s, c.lt.s, c.le.s add.d, sub.d, mul.d, div.d, abs.d, neg.d, c.eq.d, c.lt.d, c.le.d
FI-Type	l.s, s.s, l.d, s.d, bc1t, bc1f

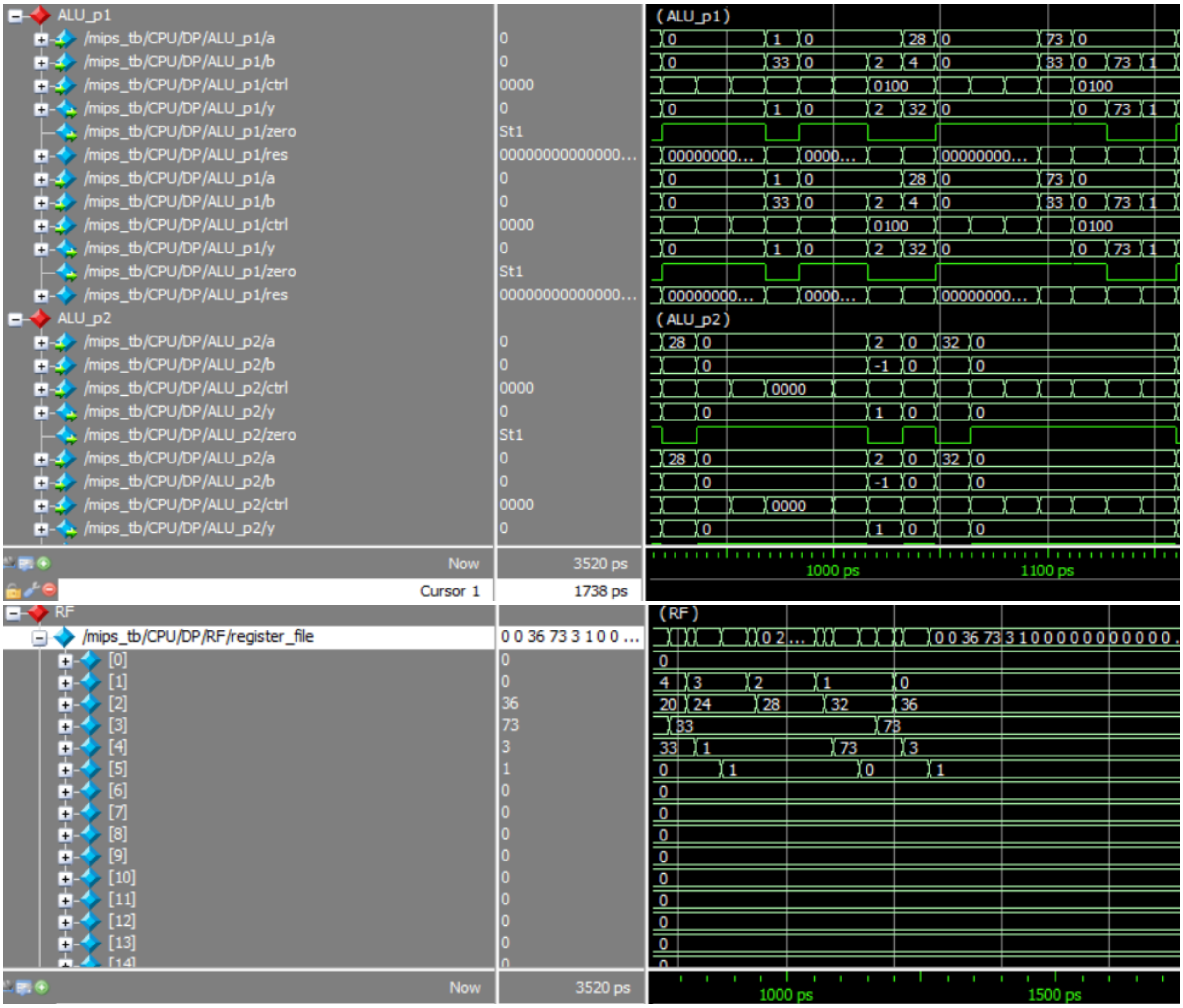
Tests:

All codes are available in folder ‘codes’. In memory section there is a text file named ‘TEST_INST.txt’ including one or more samples for each instruction and their expected results. This can be used for instructions testing. There is another file named ‘inst.txt’ which is a program that reads 10 signed words from address 1000 in memory (file ‘data.txt’) and writes their largest one in location 2000 in memory. This location will be printed at the end, like Figure 2. Macro for waves named ‘waves.do’ is saved in project folder and can be loaded to waveform window. Figure 3 shows some signals from ALU and RefisterFile units.



```
# The content of mem[2000] =          73
```

Figure 2. Maximum value in array, written in location 2000



```
ALU_p1
  /mips_tb/CPU/DP/ALU_p1/a        0
  /mips_tb/CPU/DP/ALU_p1/b        0
  /mips_tb/CPU/DP/ALU_p1/ctrl     0000
  /mips_tb/CPU/DP/ALU_p1/y        0
  /mips_tb/CPU/DP/ALU_p1/zero     St1
  /mips_tb/CPU/DP/ALU_p1/res      00000000000000...
  /mips_tb/CPU/DP/ALU_p1/a        0
  /mips_tb/CPU/DP/ALU_p1/b        0
  /mips_tb/CPU/DP/ALU_p1/ctrl     0000
  /mips_tb/CPU/DP/ALU_p1/y        0
  /mips_tb/CPU/DP/ALU_p1/zero     St1
  /mips_tb/CPU/DP/ALU_p1/res      00000000000000...
ALU_p2
  /mips_tb/CPU/DP/ALU_p2/a        0
  /mips_tb/CPU/DP/ALU_p2/b        0
  /mips_tb/CPU/DP/ALU_p2/ctrl     0000
  /mips_tb/CPU/DP/ALU_p2/y        0
  /mips_tb/CPU/DP/ALU_p2/zero     St1
  /mips_tb/CPU/DP/ALU_p2/a        0
  /mips_tb/CPU/DP/ALU_p2/b        0
  /mips_tb/CPU/DP/ALU_p2/ctrl     0000
  /mips_tb/CPU/DP/ALU_p2/y        0
Now        3520 ps
Cursor 1   1738 ps
RF
  /mips_tb/CPU/DP/RF/register_file   0 0 36 73 3 1 0 0 ...
    [0]    0
    [1]    0
    [2]    36
    [3]    73
    [4]    3
    [5]    1
    [6]    0
    [7]    0
    [8]    0
    [9]    0
    [10]   0
    [11]   0
    [12]   0
    [13]   0
    [14]   0
Now        3520 ps
```

Figure 3. ALU and RegiterFile waveforms

Summary:

This design is a 5-stage 2-way superscalar MIPS processor with limited instruction, supporting forwarding and hazard handling for implemented instructions. IF stage reads instructions, ID decodes instruction, issues them for next stages, EX does arithmetic and logical computations, MEM does data memory read and write access, and WB commits an instruction.