**Digital Design and Computer Architecture LU**

# Lab Protocol

# Exercise IV

Group 14

Philipp Geisler, Matr. Nr. 11775812

e11775812@student.tuwien.ac.at

Anna Balla, Matr. Nr. 11709469

e11709569@student.tuwien.ac.at

Vienna, August 1, 2020

# Forwarding Simulation

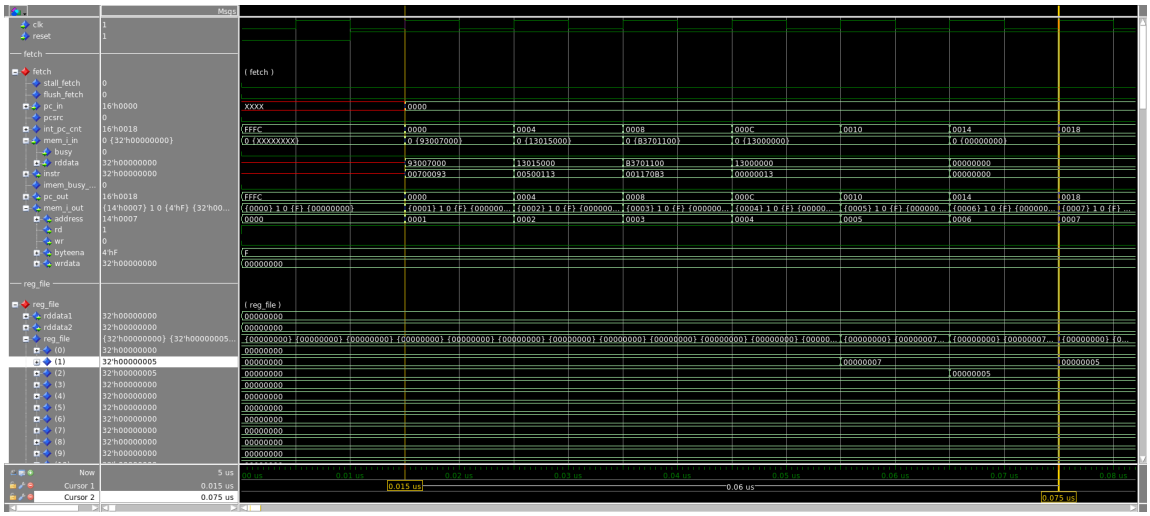

Figure 1: Simulation screenshot for Listing 1.

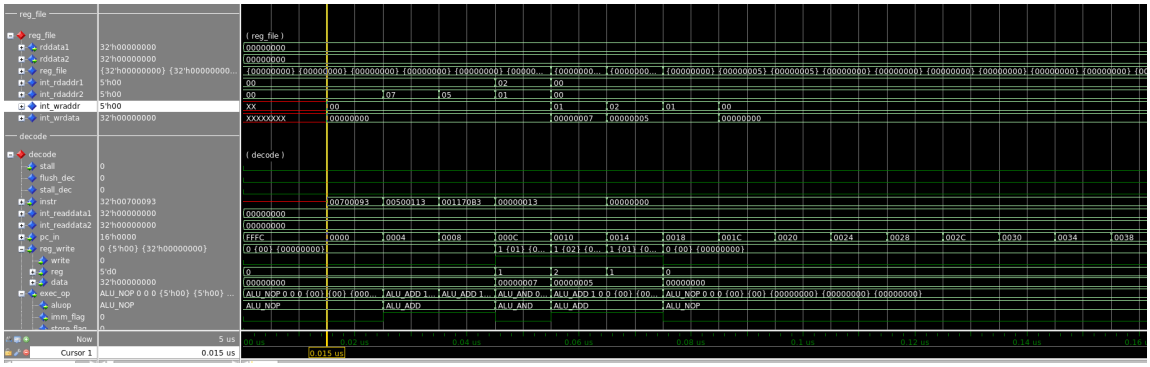

Figure 2: The internal signal wraddr and wrdata.

Unfortunately the signal `regwrite` was optimized away by the compiler. But you can see the signal that causes the regfile to write values in the decode stage. There it is the signal `reg_write.write`.

A screenshot of the exec stage is also appended.

Make sure that at least the following signals are visible in Figure 1: the program counter in the fetch stage, the instruction being fetched, the content of registers `x1` and `x2` as well as the signals `wraddr`, `wrdata` and `regwrite` of the register file.

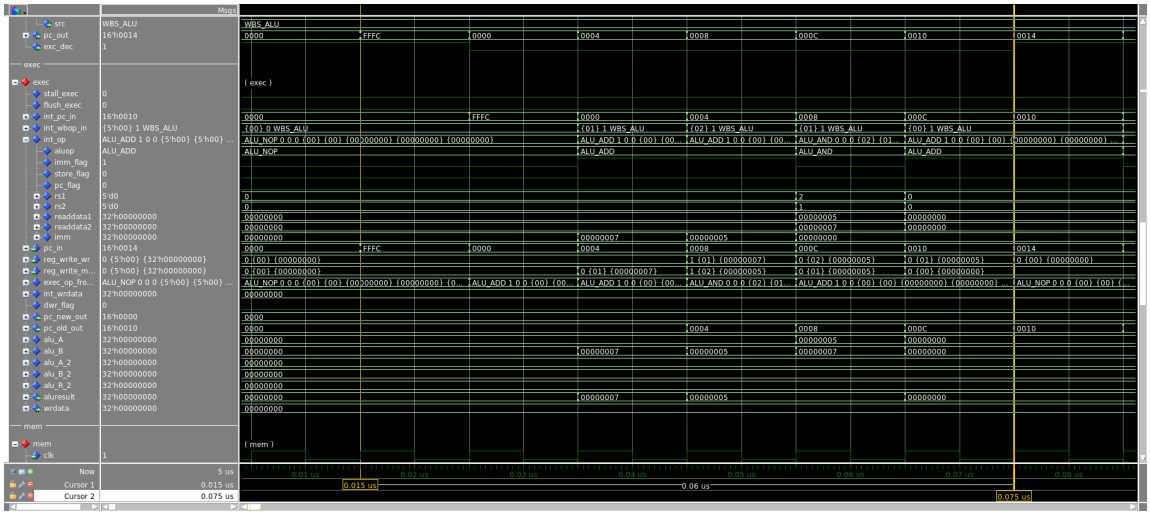

Figure 3: What's going on in the exec stage.

Listing 1: Assembler example with forwarding

```
addi x1, x0, 7
addi x2, x0, 5
and x1, x2, x1
nop
nop
```

## Branch Hazards Simulation

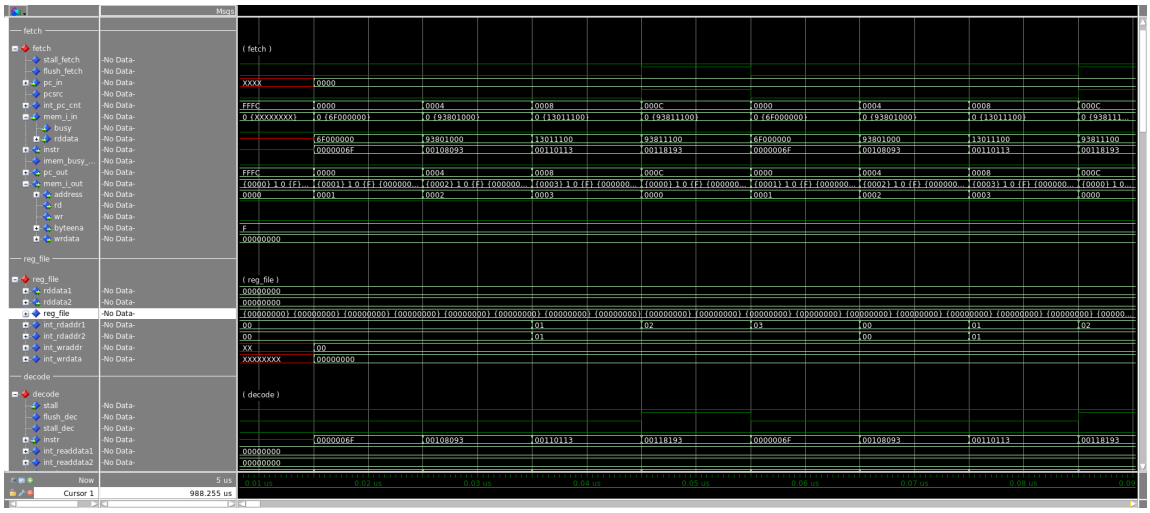

Figure 4: Simulation screenshot for Listing 2.

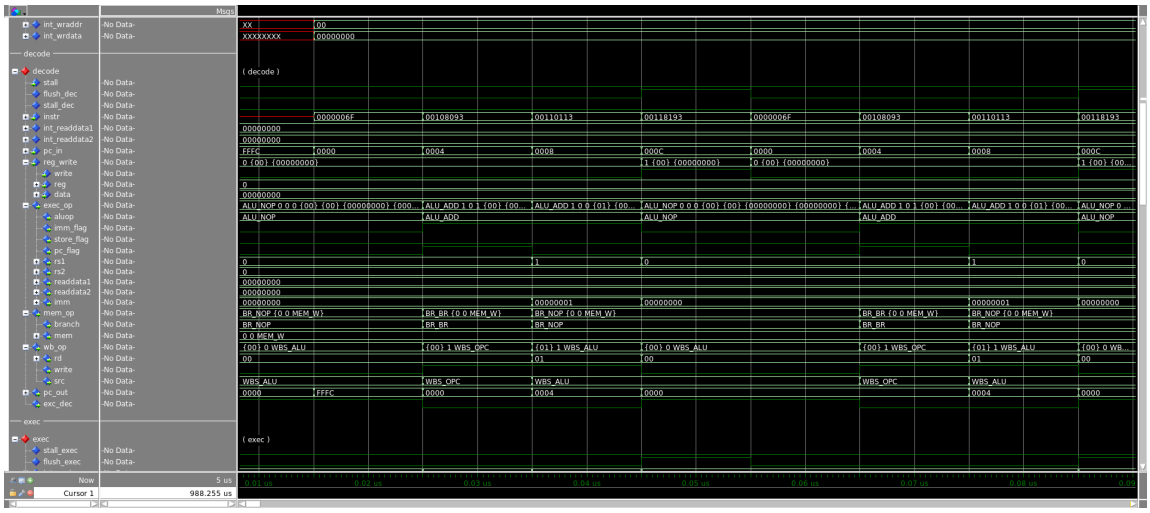

Figure 5: Simulation screenshot for Listing **??**.

Make sure that at least the following signals are visible in Figure 4: the program counter in the fetch stage, the instruction being fetched, the content of registers `x1`, `x2` and `x3` as well as the signals `wraddr`, `wrdata` and `regwrite` of the register file.

Listing 2: Assembler example for branches

```
loop:   j loop
        addi x1, x1, 1
        addi x2, x2, 1
        addi x3, x3, 1
```

## Synthesis Results

Table 1: Resource usage by entity, including resources used by sub-entities.

| | LC Combinationals | LC Registers | Memory Bits |
|---|---|---|---|
| Fetch Stage | 80 | 2 | 0 |
| Decode Stage | 253 | 2 | 0 |
| – Register File | 1132 | 99 | 0 |
| Execute Stage | 2229 | 6 | 0 |
| – ALU | 1: 517, 2:16 | 1: 0, 2: 0 | 1: 0, 2: 0 |
| Memory Stage | 154 | 6 | 0 |
| – Memory Unit | 85 | 0 | 0 |
| Write-Back Stage | 135 | 0 | 0 |
| Forwarding Unit | 1: 10, 2:5 | 1:0, 2:0 | 1: 0, 2:0 |
| Control Unit | 14 | 0 | 0 |
| Sum | 4630 | 109 | 0 |

**Question:** What is the maximum frequency of your design?

**Answer:** In the slow 1200mV 85C Model that's 80.72 MHz.

**Question:** Where is the critical path of your design?

**Answer:** That's the forwarding of values via both the forwarding units.