Digital Design and Computer Architecture LU

Lab Protocol

Exercise IV

Group ?

Klaus Botschen, Matr. Nr. 8625499

klaus.botschen@student.tuwien.ac.at

Manuel Fischer, Matr. Nr. 11905171

manuel.fischer@tuwien.ac.at

Vienna, June 28, 2021

Forwarding Simulation

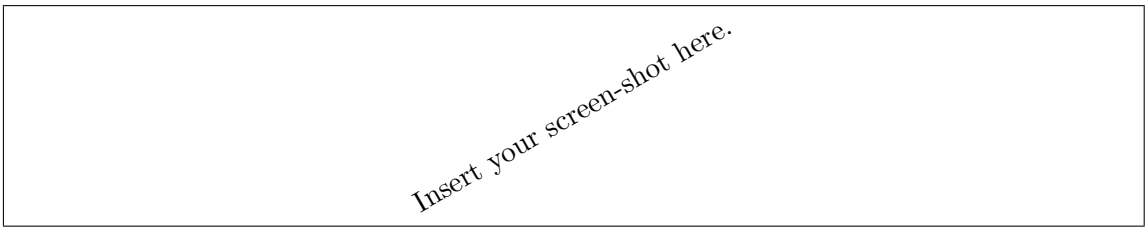

Figure 1: Simulation screenshot for Listing 1.

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.

Listing 1: Assembler example with forwarding

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

Branch Hazards Simulation

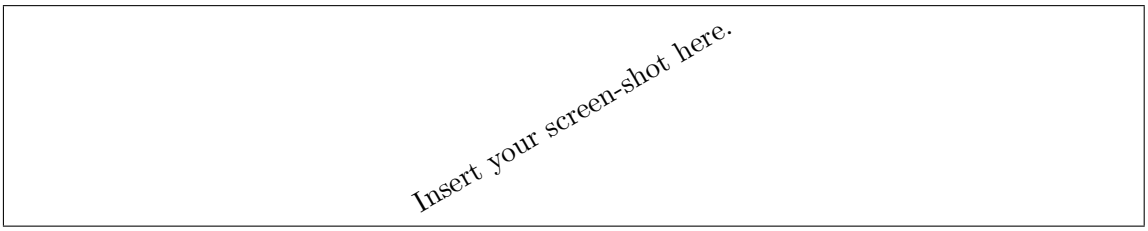

Figure 2: Simulation screenshot for Listing 2.

Make sure that at least the following signals are visible in Figure 2: 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
```

Cache Simulation

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

	LC Combinationals	LC Registers	Memory Bits
Fetch Stage	82	16	0
Decode Stage	165	47	2048
– Register File	149	162	2048
Execute Stage	336	152	0
– ALU	696	0	0
Memory Stage	88	113	0
– Memory Unit	85	0	0
Write-Back Stage	87	88	0
Forwarding Unit	15	0	0
Control Unit			
Sum			

Question: What is the maximum frequency of your design?

Answer: 92MHz with aggressive optimization at 85Â°C; with balanced mode $f_{max} = 84.36MHz$.

Question: Where is the critical path of your design?

Answer: The critical path is mostly due to the forwarding, e.g. from WB wb_next register via the EXEC forwarding mux through the ALU bit shift combinatorial logic finally to the alu_next register in the MEM stage.