TDTS08: Lab Report

Lab 3: Superscalar Processors

Name	PIN	Email
Alexander Yngve	930320-6651	aleyn573@student.liu.se
Pål Kastman	851212-7575	palka285@student.liu.se

Contents

1 Introduction

The purpose of this lab is to learn how Supersclar Processors work, and to try and modify an processor architecture to make it simpler, but it should still perform within 5% of the inital designs performance.

2 Method

We started out by investigating every part of the design individually, to see how they affected the performance of the design.

We then choose to simplify the parts that didn't affect the performance. We determined what parts we couldn't simplify due to that the performance would go further than 5% from the initial performance.

Now we looked at the parts of the design that we could modify, and at their traces.

3 Result

In order to establish a baseline performance the simulator was run with the default arguments and a trace was created, with the command shown in listing 1.

Listing 1 – Simulator command.

```
sim−outorder −config superscalar.cfg −ptrace trace.trc
    100000:+30 go.ss 3 8
```

The config file ”superscalar.cfg” is supplied with the lab and contains all default settings. The most interesting of these are shown in table 1.

Table 1 – Default settings.

Setting	value
res:ialu	4
res:imult	4
res:fpalu	2
res:fpmult	2
ruu:size	32

3.1 Integer components

3.2 Floating Point components

insert graphs of how the parameters changed

In figure **??** we can see that by changing the floating point alu & multiplier, the system didn't perform any worse, thus these parts can be simplified as much as possible.

3.3 Control components

The speed of the system was already at the lowest possible, which means by changing this value we get a more complex design, thus we decided not to.

4 Discussion

explain why go.ss doesn't need those