Tomasulo Algorithm Report

Name: Martin Lee Student Id: 1985750

Assignment problem:

Implementing the Tomasulo algorithm with support of issuing multiple instructions (1-4 instructions – in order.)

Code modification:

- Create a new variable called “issue_width”, and it can be 1-4.

```
issue_width=1
```

- And at the definition of issue in “issue.py”, I’ve added an additional parameter. Therefore, the number of issues that can be issued in a cycle is determined by the issue-width.

```
def issue(cycle, PC, instructions, ROB, size_ROB,
        rs_int_adder,
        rs_fp_adder,
        rs_fp_multi,
        ld_sd_queue, size_ld_sd_queue,
        rat_int, rat_fp, issue_width):
    for i in range(min(issue_width,len(instructions)-PC.PC)):
        # fetch issue_width instructions
```

- To test the example that given in the slide, I’ve also modified several parameters in the code, such as

	# of rs	Cycles in EX	Cycles in Mem	# of FUs
Integer adder	1	1		1
FP adder	2	4		1
FP multiplier	2	10		1
Load/store unit	2	1	6	1

- The instructions in file “test_case.in”:
  Ld F6 24(R1)
  Ld F2 32(R2)
  Mult.d F1 F2 F4
  Mult.d F8 F6 F1c
  Add.d F10 F1 F2
  Addi R2 R2 4

- I did not change the code to commit multiple instructions in each clock cycle, so the commit value of my result will be different to the result in the example slide.

Result:

Issue width 1:

	ISSUE	EXE	MEM	WB	COMMIT
Ld F6 24(R1)	[1]	[2, 2]	[3, 8]	[9]	[10]
Ld F2 32(R2)	[2]	[3, 3]	[9, 14]	[15]	[16]
Mult.d F1 F2 F4	[3]	[16, 25]	[]	[26]	[27]
Mult.d F8 F6 F1	[4]	[27, 36]	[]	[37]	[38]
Add.d F10 F1 F2	[5]	[27, 30]	[]	[31]	[39]
Addi R2 R2 4	[6]	[7, 7]	[]	[8]	[40]

Issue width 2:

	ISSUE	EXE	MEM	WB	COMMIT
Ld F6 24(R1)	[1]	[2, 2]	[3, 8]	[9]	[10]
Ld F2 32(R2)	[1]	[2, 2]	[9, 14]	[15]	[16]
Mult.d F1 F2 F4	[2]	[16, 25]	[]	[26]	[27]
Mult.d F8 F6 F1	[2]	[27, 36]	[]	[37]	[38]
Add.d F10 F1 F2	[3]	[27, 30]	[]	[31]	[39]
Addi R2 R2 4	[3]	[4, 4]	[]	[5]	[40]

Issue width 3:

	ISSUE	EXE	MEM	WB	COMMIT
Ld F6 24(R1)	[1]	[2, 2]	[3, 8]	[9]	[10]
Ld F2 32(R2)	[1]	[2, 2]	[9, 14]	[15]	[16]
Mult.d F1 F2 F4	[1]	[16, 25]	[]	[26]	[27]
Mult.d F8 F6 F1	[2]	[27, 36]	[]	[37]	[38]
Add.d F10 F1 F2	[2]	[27, 30]	[]	[31]	[39]
Addi R2 R2 4	[2]	[3, 3]	[]	[4]	[40]

Issue width 4:

	ISSUE	EXE	MEM	WB	COMMIT
Ld F6 24(R1)	[1]	[2, 2]	[3, 8]	[9]	[10]
Ld F2 32(R2)	[1]	[2, 2]	[9, 14]	[15]	[16]
Mult.d F1 F2 F4	[1]	[16, 25]	[]	[26]	[27]
Mult.d F8 F6 F1	[1]	[27, 36]	[]	[37]	[38]
Add.d F10 F1 F2	[2]	[27, 30]	[]	[31]	[39]
Addi R2 R2 4	[2]	[3, 3]	[]	[4]	[40]