ECE4700J Homework 3

Yiwen Yang

Q1

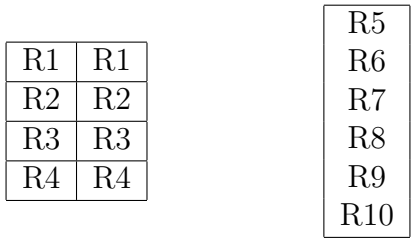
1. 16 bypass paths, 16 point-to-point connections, 4 5x1 muxes.
2. 8 bypass paths, 8 point-to-point connections, 4 3x1 muxes.

Q2

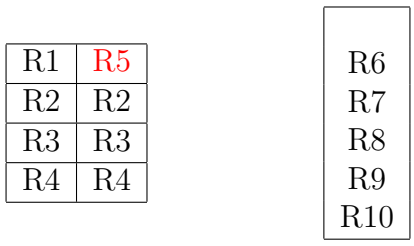
1.

write-read	write-write	read-write
L2-L3	L1-L2	L2-L4
L3-L4	L2-L5	L1-L4
L4-L5	L1-L5	L2-L5
L2-L4		L1-L5

2.



`R1 = 100` → `R5 = 100` [`R1`]



```
    R1 = 100 → R5 = 100 [R1]
R1 = R2 + R4 → R6 = R2 + R4 [R1]
```

R1	R6
R2	R2
R3	R3
R4	R4

R7
R8
R9
R10

```
    R1 = 100 → R5 = 100 [R1]
R1 = R2 + R4 → R6 = R2 + R4 [R1]
R2 = R4 - 25 → R7 = R4 - 25 [R2]
```

R1	R6
R2	R7
R3	R3
R4	R4

R8
R9
R10

```
    R1 = 100 → R5 = 100 [R1]
R1 = R2 + R4 → R6 = R2 + R4 [R1]
R2 = R4 - 25 → R7 = R4 - 25 [R2]
R4 = R1 + R3 → R8 = R6 + R3 [R4]
```

R1	R6
R2	R7
R3	R3
R4	R8

R9
R10

```
    R1 = 100 → R5 = 100 [R1]
R1 = R2 + R4 → R6 = R2 + R4 [R1]
R2 = R4 - 25 → R7 = R4 - 25 [R2]
R4 = R1 + R3 → R8 = R6 + R3 [R4]
R1 = R1 + 30 → R9 = R6 + 30 [R1]
```

R1	R9
R2	R7
R3	R3
R4	R8

R10

Q3

1. Compilers are difficult to design.
2. Register file bandwidth and memory bandwidth will be large.

Q4

1. Cycle 0

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F											
`mult r3*r2 → r4`	F											
`add r2+r5 → r4`												
`or r4|r6 → r7`												
`sub r7-3 → r8`												
`ld [r9] → r7`												

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p9	p2	p5	p1	p4	p3	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	–	–	–	–	–	–

Instruction	To Free	Done
ld		no
mult		no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday

2. Cycle 1a

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di										
`mult r3*r2 → r4`	F											
`add r2+r5 → r4`												
`or r4|r6 → r7`												
`sub r7-3 → r8`												
`ld [r9] → r7`												

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p2	p5	p1	p4	p3	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	no	–	–	–	–	–

Instruction	To Free	Done
ld	p9	no
mult		no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0

3. Cycle 1b

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di										
`mult r3*r2 → r4`	F	Di										
`add r2+r5 → r4`												
`or r4|r6 → r7`												
`sub r7-3 → r8`												
`ld [r9] → r7`												

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p11	p5	p1	p4	p3	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	no	no	–	–	–	–

Instruction	To Free	Done
ld	p9	no
mult	p2	no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	no	p7	yes	p11	1

4. Cycle 1c

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di										
`mult r3*r2 → r4`	F	Di										
`add r2+r5 → r4`		F										
`or r4|r6 → r7`		F										
`sub r7-3 → r8`												
`ld [r9] → r7`												

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p11	p5	p1	p4	p3	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	no	no	–	–	–	–

Instruction	To Free	Done
ld	p9	no
mult	p2	no
add		no
or		no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	no	p7	yes	p11	1

5. Cycle 2a

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I									
`mult r3*r2 → r4`	F	Di										
`add r2+r5 → r4`		F										
`or r4|r6 → r7`		F										
`sub r7-3 → r8`												
`ld [r9] → r7`												

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p11	p5	p1	p4	p3	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	no	no	–	–	–	–

Instruction	To Free	Done
ld	p9	no
mult	p2	no
add		no
or		no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	no	p7	yes	p11	1

6. Cycle 2b

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I									
`mult r3*r2 → r4`	F	Di										
`add r2+r5 → r4`		F	Di									
`or r4|r6 → r7`		F										
`sub r7-3 → r8`												
`ld [r9] → r7`												

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p4	p3	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	no	no	no	–	–	–

Instruction	To Free	Done
ld	p9	no
mult	p2	no
add	p11	no
or		no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	no	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2

7. Cycle 2c

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I									
`mult r3*r2 → r4`	F	Di										
`add r2+r5 → r4`		F	Di									
`or r4|r6 → r7`		F	Di									
`sub r7-3 → r8`												
`ld [r9] → r7`												

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p13	p3	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	no	no	no	no	–	–

Instruction	To Free	Done
ld	p9	no
mult	p2	no
add	p11	no
or	p4	no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	no	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2
or	p12	no	p1	yes	p13	3

8. Cycle 2d

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I									
`mult r3*r2 → r4`	F	Di										
`add r2+r5 → r4`		F	Di									
`or r4|r6 → r7`		F	Di									
`sub r7-3 → r8`			F									
`ld [r9] → r7`			F									

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p13	p3	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	no	no	no	no	–	–

Instruction	To Free	Done
ld	p9	no
mult	p2	no
add	p11	no
or	p4	no
sub		no
ld		no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	no	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2
or	p12	no	p1	yes	p13	3

9. Cycle 3a

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I	RR								
`mult r3*r2 → r4`	F	Di										
`add r2+r5 → r4`		F	Di	I								
`or r4|r6 → r7`		F	Di									
`sub r7-3 → r8`			F									
`ld [r9] → r7`			F									

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p13	p3	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	no	no	no	no	–	–

Instruction	To Free	Done
ld	p9	no
mult	p2	no
add	p11	no
or	p4	no
sub		no
ld		no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	no	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2
or	p12	no	p1	yes	p13	3

10. Cycle 3b

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I	RR								
`mult r3*r2 → r4`	F	Di										
`add r2+r5 → r4`		F	Di	I								
`or r4|r6 → r7`		F	Di									
`sub r7-3 → r8`			F	Di								
`ld [r9] → r7`			F									

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p13	p14	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	no	no	no	no	no	–

Instruction	To Free	Done
ld	p9	no
mult	p2	no
add	p11	no
or	p4	no
sub	p3	no
ld		no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	no	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2
or	p12	no	p1	yes	p13	3
sub	p13	no	–	yes	p14	4

11. Cycle 3c

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I	RR								
`mult r3*r2 → r4`	F	Di										
`add r2+r5 → r4`		F	Di	I								
`or r4|r6 → r7`		F	Di									
`sub r7-3 → r8`			F	Di								
`ld [r9] → r7`			F	Di								

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p15	p14	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	no	no	no	no	no	no

Instruction	To Free	Done
ld	p9	no
mult	p2	no
add	p11	no
or	p4	no
sub	p3	no
ld	p13	no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	no	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2
or	p12	no	p1	yes	p13	3
sub	p13	no	–	yes	p14	4
ld	p6	yes	–	yes	p15	5

12. Cycle 3d

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I	RR								
`mult r3*r2 → r4`	F	Di										
`add r2+r5 → r4`		F	Di	I								
`or r4|r6 → r7`		F	Di									
`sub r7-3 → r8`			F	Di								
`ld [r9] → r7`			F	Di								

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p15	p14	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	no	no	yes	no	no	no

Instruction	To Free	Done
ld	p9	no
mult	p2	no
add	p11	no
or	p4	no
sub	p3	no
ld	p13	no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	no	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2
or	p12	yes	p1	yes	p13	3
sub	p13	no	–	yes	p14	4
ld	p6	yes	–	yes	p15	5

13. Cycle 4a

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I	RR								
`mult r3*r2 → r4`	F	Di										
`add r2+r5 → r4`		F	Di	I								
`or r4|r6 → r7`		F	Di		I							
`sub r7-3 → r8`			F	Di								
`ld [r9] → r7`			F	Di	I							

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p15	p14	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	no	no	yes	no	no	no

Instruction	To Free	Done
ld	p9	no
mult	p2	no
add	p11	no
or	p4	no
sub	p3	no
ld	p13	no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	no	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2
or	p12	yes	p1	yes	p13	3
sub	p13	no	–	yes	p14	4
ld	p6	yes	–	yes	p15	5

14. Cycle 4b

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I	RR	X							
`mult r3*r2 → r4`	F	Di										
`add r2+r5 → r4`		F	Di	I	RR							
`or r4|r6 → r7`		F	Di		I							
`sub r7-3 → r8`			F	Di								
`ld [r9] → r7`			F	Di	I							

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p15	p14	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	no	yes	yes	no	no

Instruction	To Free	Done
ld	p9	no
mult	p2	no
add	p11	no
or	p4	no
sub	p3	no
ld	p13	no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	yes	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2
or	p12	yes	p1	yes	p13	3
sub	p13	yes	–	yes	p14	4
ld	p6	yes	–	yes	p15	5

15. Cycle 5a

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I	RR	X							
`mult r3*r2 → r4`	F	Di				I						
`add r2+r5 → r4`		F	Di	I	RR							
`or r4|r6 → r7`		F	Di		I							
`sub r7-3 → r8`			F	Di		I						
`ld [r9] → r7`			F	Di	I							

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p15	p14	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	no	yes	yes	yes	no

Instruction	To Free	Done
ld	p9	no
mult	p2	no
add	p11	no
or	p4	no
sub	p3	no
ld	p13	no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	yes	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2
or	p12	yes	p1	yes	p13	3
sub	p13	yes	–	yes	p14	4
ld	p6	yes	–	yes	p15	5

16. Cycle 5b

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I	RR	X	M1						
`mult r3*r2 → r4`	F	Di				I						
`add r2+r5 → r4`		F	Di	I	RR	X						
`or r4|r6 → r7`		F	Di		I	RR						
`sub r7-3 → r8`			F	Di		I						
`ld [r9] → r7`			F	Di	I	RR						

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p15	p14	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	no	yes	yes	yes	no

Instruction	To Free	Done
ld	p9	no
mult	p2	no
add	p11	no
or	p4	no
sub	p3	no
ld	p13	no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	yes	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2
or	p12	yes	p1	yes	p13	3
sub	p13	yes	–	yes	p14	4
ld	p6	yes	–	yes	p15	5

17. Cycle 6

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I	RR	X	M1	M2					
`mult r3*r2 → r4`	F	Di				I	RR					
`add r2+r5 → r4`		F	Di	I	RR	X	W					
`or r4|r6 → r7`		F	Di		I	RR	X					
`sub r7-3 → r8`			F	Di		I	RR					
`ld [r9] → r7`			F	Di	I	RR	X					

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p15	p14	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	no	yes	yes	yes	yes

Instruction	To Free	Done
ld	p9	no
mult	p2	no
add	p11	yes
or	p4	no
sub	p3	no
ld	p13	no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	yes	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2
or	p12	yes	p1	yes	p13	3
sub	p13	yes	–	yes	p14	4
ld	p6	yes	–	yes	p15	5

18. Cycle 7

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I	RR	X	M1	M2	W				
`mult r3*r2 → r4`	F	Di				I	RR	X1				
`add r2+r5 → r4`		F	Di	I	RR	X	W	C				
`or r4|r6 → r7`		F	Di		I	RR	X	W				
`sub r7-3 → r8`			F	Di		I	RR	X				
`ld [r9] → r7`			F	Di	I	RR	X	M1				

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p15	p14	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	–	yes	yes	yes	yes

Instruction	To Free	Done
ld	p9	yes
mult	p2	no
add	p11	yes
or	p4	yes
sub	p3	no
ld	p13	no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	yes	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2
or	p12	yes	p1	yes	p13	3
sub	p13	yes	–	yes	p14	4
ld	p6	yes	–	yes	p15	5

19. Cycle 8

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I	RR	X	M1	M2	W	C			
`mult r3*r2 → r4`	F	Di				I	RR	X1	X2			
`add r2+r5 → r4`		F	Di	I	RR	X	W	C				
`or r4|r6 → r7`		F	Di		I	RR	X	W	C			
`sub r7-3 → r8`			F	Di		I	RR	X	W			
`ld [r9] → r7`			F	Di	I	RR	X	M1	M2			

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p15	p14	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	yes	–	yes	yes	yes	yes	–	yes	–	yes	yes	yes	yes

Instruction	To Free	Done
ld	p9	yes
mult	p2	no
add	p11	yes
or	p4	yes
sub	p3	yes
ld	p13	no

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	yes	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2
or	p12	yes	p1	yes	p13	3
sub	p13	yes	–	yes	p14	4
ld	p6	yes	–	yes	p15	5

20. Cycle 9

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I	RR	X	M1	M2	W	C			
`mult r3*r2 → r4`	F	Di				I	RR	X1	X2	X3		
`add r2+r5 → r4`		F	Di	I	RR	X	W	C				
`or r4|r6 → r7`		F	Di		I	RR	X	W	C			
`sub r7-3 → r8`			F	Di		I	RR	X	W	C		
`ld [r9] → r7`			F	Di	I	RR	X	M1	M2	W		

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p15	p14	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	–	–	yes	yes	yes	yes	–	yes	–	yes	yes	yes	yes

Instruction	To Free	Done
ld	p9	yes
mult	p2	no
add	p11	yes
or	p4	yes
sub	p3	yes
ld	p13	yes

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	yes	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2
or	p12	yes	p1	yes	p13	3
sub	p13	yes	–	yes	p14	4
ld	p6	yes	–	yes	p15	5

21. Cycle 10

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I	RR	X	M1	M2	W	C			
`mult r3*r2 → r4`	F	Di				I	RR	X1	X2	X3	W	
`add r2+r5 → r4`		F	Di	I	RR	X	W	C				
`or r4|r6 → r7`		F	Di		I	RR	X	W	C			
`sub r7-3 → r8`			F	Di		I	RR	X	W	C		
`ld [r9] → r7`			F	Di	I	RR	X	M1	M2	W	C	

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p15	p14	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	yes	–	–	yes	yes	yes	yes	–	yes	–	yes	–	yes	yes

Instruction	To Free	Done
ld	p9	yes
mult	p2	yes
add	p11	yes
or	p4	yes
sub	p3	yes
ld	p13	yes

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	yes	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2
or	p12	yes	p1	yes	p13	3
sub	p13	yes	–	yes	p14	4
ld	p6	yes	–	yes	p15	5

22. Cycle 11 (Done)

	0	1	2	3	4	5	6	7	8	9	10	11
`ld [r1] → r3`	F	Di	I	RR	X	M1	M2	W	C			
`mult r3*r2 → r4`	F	Di				I	RR	X1	X2	X3	W	C
`add r2+r5 → r4`		F	Di	I	RR	X	W	C				
`or r4|r6 → r7`		F	Di		I	RR	X	W	C			
`sub r7-3 → r8`			F	Di		I	RR	X	W	C		
`ld [r9] → r7`			F	Di	I	RR	X	M1	M2	W	C	

r1	r2	r3	r4	r5	r6	r7	r8	r9
p8	p7	p10	p12	p5	p1	p15	p14	p6

p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15
yes	–	–	–	yes	yes	yes	yes	–	yes	–	yes	–	yes	yes

Instruction	To Free	Done
ld	p9	yes
mult	p2	yes
add	p11	yes
or	p4	yes
sub	p3	yes
ld	p13	yes

Instruction	rs1	Ready	rs2	Ready	rd	Birthday
ld	p8	yes	–	yes	p10	0
mult	p10	yes	p7	yes	p11	1
add	p7	yes	p5	yes	p12	2
or	p12	yes	p1	yes	p13	3
sub	p13	yes	–	yes	p14	4
ld	p6	yes	–	yes	p15	5

Q5

1. If x2 and x8 all have some kind of dependencies before these instructions, such as branches or register writebacks, and x3 does not have data dependencies, then instruction 5 may be executed first.

2. This is not possible, since instruction 3 and 4 has a RAW dependency on x6, then instruction 4 has to be anyway executed after instruction 3.

OQ1

I would suggest using a 1-bit predictor for this branch. Since this branch shows a pattern of continually taking either T or N in a certain period, then a 1-bit predictor can switch the prediction faster than the 2-bit one so there will be fewer miss predictions.