

# Maintaining Precise State: Reorder Buffer

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
① L.D F0, A(R1)	F	D	I	E	M	W	C																							
② L.D F2, B(R1)	F	D	I	E	M	W	C																							
③ MUL.D F4, F0, F2	F	D	I	E	M	W	C																							
④ ADD.D F6, F4, F8	F	D	I	E	M	W	C																							
⑤ S.D F6, D(R1)	F	D	I	E	M	W	C																							
⑥ DADDI R1, R1, #8	F	D	I	E	M	W	C																							
⑦ SLT R3, R1, R2	F	D	I	E	M	W	C																							
⑧ BNEZ R3, LOOP	F	D	I	E	M	W	C																							
⑨ L.D F0, A(R1)	F	D	I	E	M	W	C																							
⑩ L.D F2, B(R1)	F	D	I	E	M	W	C																							
⑪ MUL.D F4, F0, F2	F	D	I	E	M	W	C																							
⑫ ADD.D F6, F4, F8	F	D	I	E	M	W	C																							
⑬ S.D F6, D(R1)	F	D	I	E	M	W	C																							
⑭ DADDI R1, R1, #8	F	D	I	E	M	W	C																							
⑮ SLT R3, R1, R2	F	D	I	E	M	W	C																							
⑯ BNEZ R3, LOOP	F	D	I	E	M	W	C																							

Out-of-order  
Completion  
In-order  
Commit

MUL.D - 2 cycles  
ADD.D - 2 cycles

```
for (i=0; i<n; i++) {
    D[i] = A[i] * B[i] + C;
}
```

i → R1, C → F8, n.8 → R2

iteration i

iteration i+1

# Control Speculation through Dynamic Branch Prediction

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
① L.D F0, A(R1)	F	D	I	E	M	W	C																							
② L.D F2, B(R1)	F	D	I	E	M	W	C																							
③ MUL.D F4, F0, F2	F	D	I	E	M	W	C																							
④ ADD.D F6, F4, F8	F	D	I	E	M	W	C																							
⑤ S.D F6, D(R1)	F	D	I	E	M	W	C																							
⑥ DADDI R1, R1, #8	F	D	I	E	M	W	C																							
⑦ SLT R3, R1, R2	F	D	I	E	M	W	C																							
⑧ BNEZ R3, LOOP	F	D	I	E	M	W	C																							
⑨ L.D F0, A(R1)	F	D	I	E	M	W	C																							
⑩ L.D F2, B(R1)	F	D	I	E	M	W	C																							
⑪ MUL.D F4, F0, F2	F	D	I	E	M	W	C																							
⑫ ADD.D F6, F4, F8	F	D	I	E	M	W	C																							
⑬ S.D F6, D(R1)	F	D	I	E	M	W	C																							
⑭ DADDI R1, R1, #8	F	D	I	E	M	W	C																							
⑮ SLT R3, R1, R2	F	D	I	E	M	W	C																							
⑯ BNEZ R3, LOOP	F	D	I	E	M	W	C																							
⑰ L.D F0, A(R1)	F	D	I	E	M	W	C																							
⑱ L.D F2, B(R1)	F	D	I	E	M	W	C																							

Instructions  
in reorder buffer

iteration 0

iteration 1

for (i=0; i < n; i++) {  
 D[i] = A[i] \* B[i] + C;  
}

MUL.D - 1 cycle  
 ADD.D - 1 cycle

# Control Speculation: Branch Misprediction

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Iteration N-1	①	L.D F0, A(R1)	F	D	I	E	M	W	C	:																				
	②	L.D F2, B(R1)	F	D	I	E	M	W	C	:																				
	③	MUL.D F4, F0, F2	F	D	I	I	-	E	E	W	C																			
	④	ADD.D F6, F4, F8	F	D	I	I	-	-	E	E	W	C																		
	⑤	S.D F6, D(R1)	F	D	I	(E)	-	-	E	M	W	C																		
	⑥	DADDI R1, R1, #8	F	D	I	E	M	W	-	-	-	C																		
	⑦	SLT R3, R1, R2	F	D	I	E	M	W	-	-	-	C																		
	⑧	BNEZ R3, LOOP	F	D	I	I	-	E	-	-	-	N																		
Iteration N	⑨	L.D F0, A(R1)	F	D	I	E	-	-																						
	⑩	L.D F2, B(R1)	F	D	I	E	-	-																						
	⑪	MUL.D F4, F0, F2	F	D	I	I	-	-																						
	⑫	ADD.D F6, F4, F8	F	D	I	I	-	-																						
	⑬	S.D F6, D(R1)	F	D	I	I	-	-																						
	⑭	DADDI R1, R1, #8	F	D	I	I	-	-																						
	⑮	SLT R3, R1, R2	F	D	I	I	-	-																						
	⑯	BNEZ R3, LOOP	F	D	I	I	-	-																						
⑰	Fall Through 1	F	D	I	E	M	W	C																						
⑱	Fall Through 2	F	D	I	E	M	W	C																						
⑲	Fall Through 3	F	D	I	E	M	W	C																						
⑳	Fall Through 4	F	D	I	E	M	W	C																						

instructions  
SQUASHED

