EXAMPLE: SCOREBOARDING (in-order issue, out-of-order completion)

i1:	R4<-R0*R2	
i2:	R6<-R4*R8	RAW with i1
i3:	R8<-R2+R12	WAR with i2
i4:	R4<-R14+R16	WAW with i1

Pipeline architecture

mult1	six cycle latency
mult2	six cycle latency
add	one cycle latency
shift	one cycle latency

cycle 1

instruction status

i1	issue
i2	
i3	
i4	

functional unit status

	Ua	op	Fi	Fj	Fk	Qj	Qk	Rj	Rk
shift	0								
mult1	1	i1	R4	R0	R2	?	?	1	1
mult2	0								
add	0								

register result status

R0	R2	R4	R6	R8	R10	R12	R14	R16
0	0	mult1	0	0	0	0	0	0

unit mult1 is initially free and $R_4=0$ ==> i1 can be issued

both are = 1: no RAW for i1, which can be dispatched Moreover $R_8=0$ ==> i2 can be issued

cycle 2

instruction status

i1	dispatch → read opd.
i2	issue
i3	
i4	

functional unit status

	Ua	op	Fi	Fj	Fk	Qj	Qk	Rj	Rk
shift	0								
mult1	1	i1	R4	R0	R2	?	?	1	1
mult2	1	i2	R6	R4	R8	mult1	?	0	1
add	0								

register result status

R0	R2	R4	R6	R8	R10	R12	R14	R16
0	0	mult1	mult2	0	0	0	0	0

RAW for i2, which cannot be dispatched;
$R_8=0$ => i3 is issued
i1 starts execution

cycle 3

instruction status

i1	exec(1)
i2	issue
i3	issue
i4	

functional unit status

	Ua	op	Fi	Fj	Fk	Qj	Qk	Rj	Rk
shift	0								
mult1	1	i1	R4	R0	R2	?	?	1	1
mult2	1	i2	R6	R4	R8	mult1	?	0	1
add	1	i3	R8	R2	R12	?	?	1	1

register result status

R0	R2	R4	R6	R8	R10	R12	R14	R16
0	0	mult1	mult2	add	0	0	0	0

i1 continues exec.
i2 is still stalled, as $R_j=0$ (RAW
i3 is dispatched
i4 cannot be issued for two reasons:
1) $U_A \neq 0$ (no unit free)
2) $R \neq 0$ (WAW)

cycle 4

instruction status		
	i1	exec(2)
	i2	issue
	i3	dispatch
	i4	

functional unit status	Ua	op	Fi	Fj	Fk	Qj	Qk	Rj	Rk
shift	0								
mult1	1	i1	R4	R0	R2	?	?	1	1
mult2	1	i2	R6	R4	R8	mult1	?	0	1
add	1	i3	R8	R2	R12	?	?	1	1

register result status	R0	R2	R4	R6	R8	R10	R12	R14	R16
	0	0	mult1	mult2	add	0	0	0	0

i1 continues exec.
i2 is stalled
i3 is executed
i4 blocked

cycle 5

instruction status		
	i1	exec(3)
	i2	issue
	i3	exec
	i4	

functional unit status	Ua	op	Fi	Fj	Fk	Qj	Qk	Rj	Rk
shift	0								
mult1	1	i1	R4	R0	R2	?	?	1	1
mult2	1	i2	R6	R4	R8	mult1	?	0	1
add	1	i3	R8	R2	R12	?	?	1	1

register result status	R0	R2	R4	R6	R8	R10	R12	R14	R16
	0	0	mult1	mult2	add	0	0	0	0

cycle 6	no change
cycle 7	no change
cycle 8	no change

i1 continues exec.
i2 is stalled
i3 cannot write because
$F_K(i2) = F_i(i3) \Rightarrow$ WAR

cycle 9

instruction status		
	i1	write
	i2	dispatch
	i3	write
	i4	issue

functional unit status	Ua	op	Fi	Fj	Fk	Qj	Qk	Rj	Rk
shift	0								
mult1	0								
mult2	1	i2	R6	R4	R8	mult1	?	1	1
add	1	i4	R4	R14	R16	?	?	1	1

register result status	R0	R2	R4	R6	R8	R10	R12	R14	R16
	0	0	add	mult2	0	0	0	0	0

After cycle 8:

i1 completes exec and starts write
- mult1 is cleared
- $R_j(i2)$ is set to 1

i2 is dispatched
i3 starts write
- add is replaced with i4
- R_8 is reset

i4 is issued