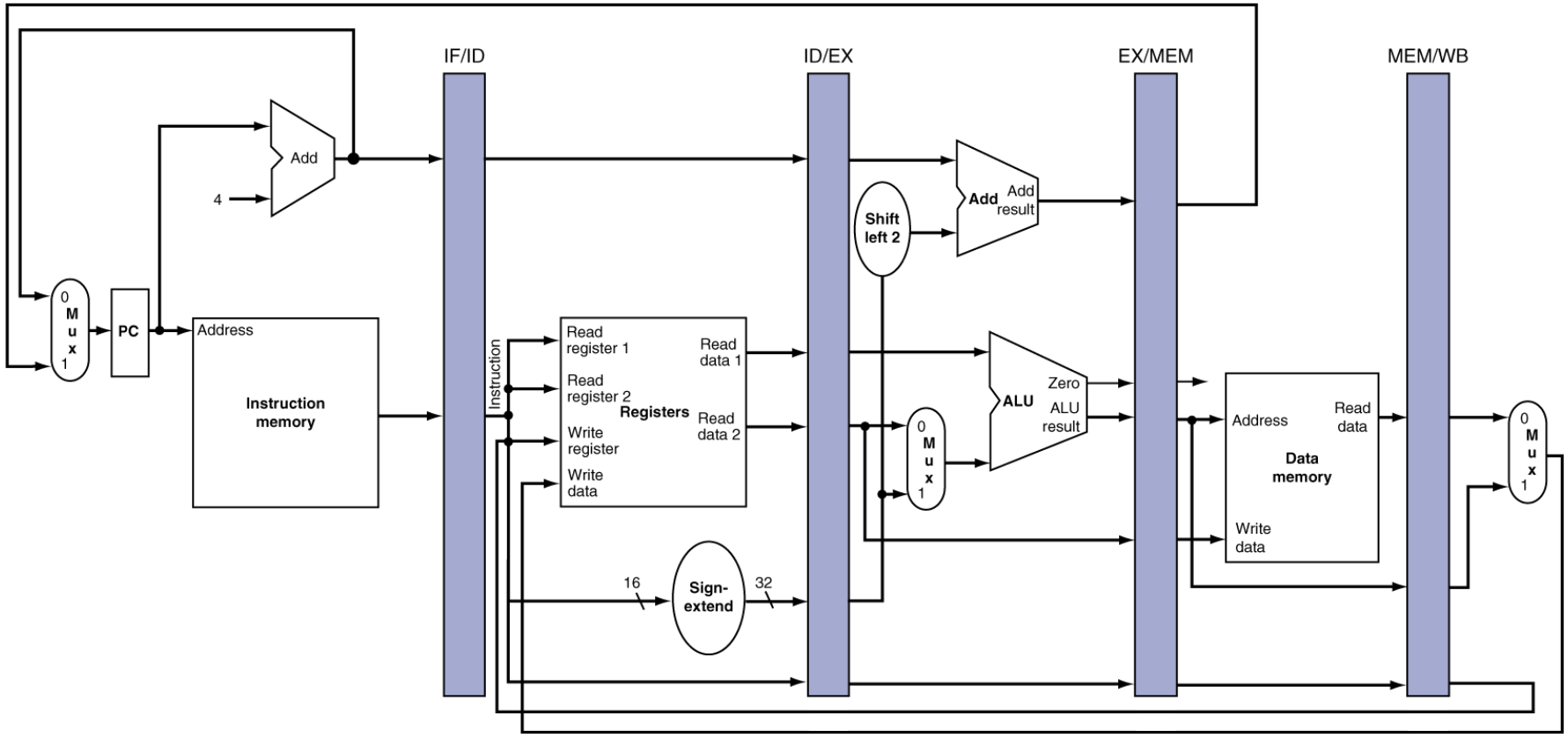
Pipeline timeline

IF	ID	EX	MEM	WB	√						
	IF	ID	EX	MEM	WB	√					
		IF	ID	EX	MEM	WB	√				
			IF	ID	EX	MEM	WB	√			
				IF	ID	EX	MEM	WB	√		
					IF	ID	EX	MEM	WB	√	
						IF	ID	EX	MEM	WB	√

Without pipelining

IF	ID	EX	MEM	WB	√									
					IF	ID	EX	MEM	WB	√				
										IF	ID	EX		

Buffering between stages



Structural hazard

IF	ID	EX	**MEM**	WB	√						
	IF	ID	EX	**MEM**	WB	√					
		IF	ID	EX	**MEM**	WB	√				
			IF	ID	EX	**MEM**	WB	√			
				IF	ID	EX	MEM	WB	√		
					IF	ID	EX	MEM	WB	√	
						IF	ID	EX	MEM	WB	√

Not all ops lw / sw

IF	ID	EX	**MEM**	**WB**	√					
	IF	ID	EX	**WB**	√					
		IF	ID	EX	WB	√				
			IF	ID	EX	WB	√			
				IF	ID	EX	WB	√		
					IF	ID	EX	WB	√	
						IF	ID	EX	WB	√

- In preceding slide:
  - Deleting inapplicable MEM stages for instructions that don’t use memory made sense, but then had resource contention for WB stage…

Not all ops lw / sw

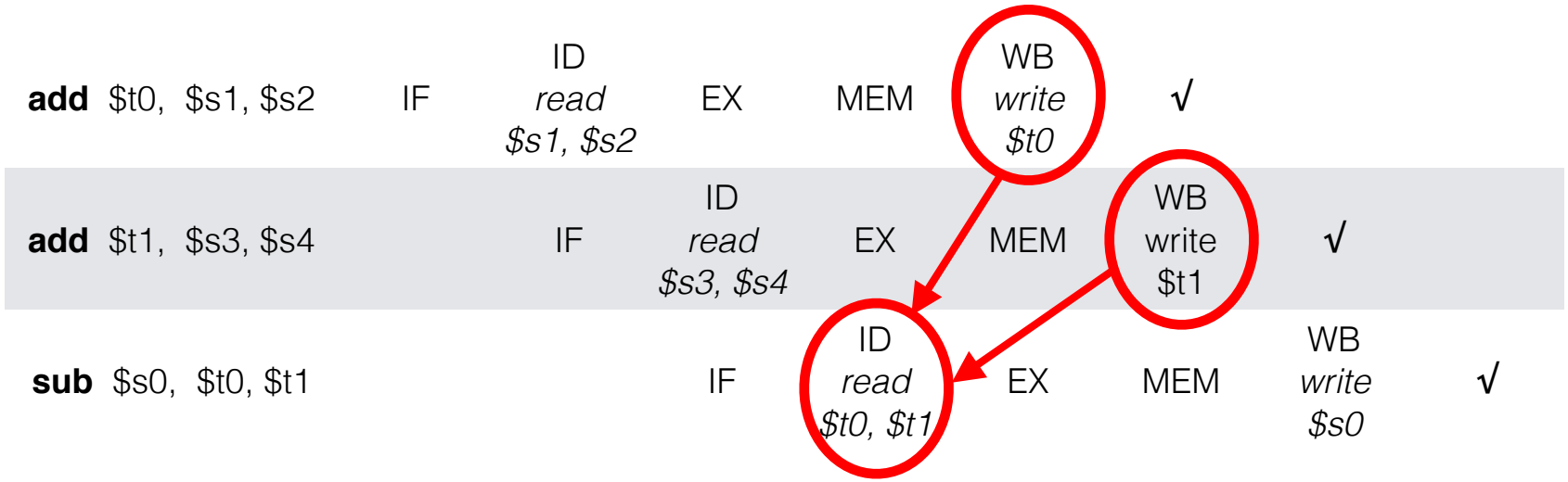
IF	ID	EX	**MEM**	WB	√						
	IF	ID	EX		WB	√					
		IF	ID	EX		WB	√				
			IF	ID	EX		WB	√			
				IF	ID	EX		WB	√		
					IF	ID	EX		WB	√	
						IF	ID	EX		WB	√

- In preceding slide:
  - In reality, we just leave these gaps so everything stays in lock step

“Bubble”

IF	ID	EX	**MEM**	WB	√			***(no √)***				
	IF	ID	EX		WB	√						
		IF	ID	EX		WB	√					
			(WAIT)	*IF*	ID	EX		WB	√			
					IF	ID	EX		WB	√		
						IF	ID	EX		WB	√	
							IF	ID	EX		WB	√

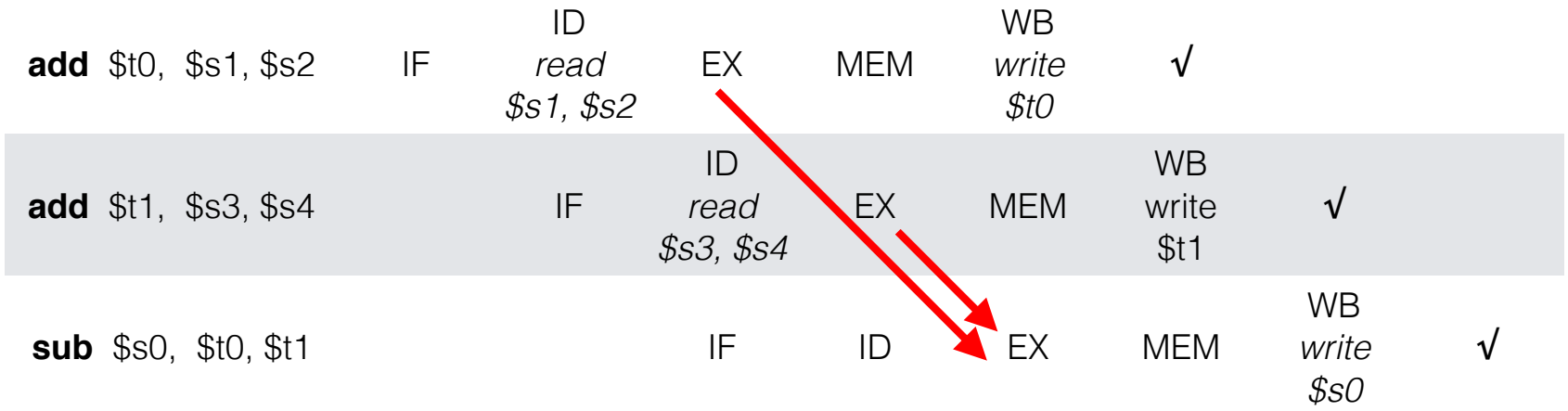
Data hazard


add $t0, $s1, $s2
IF
ID
read
$s1, $s2
EX
MEM
WB
write
$t0
√
add $t1, $s3, $s4
IF
ID
read
$s3, $s4
EX
MEM
WB
write
$t1
√
sub $s0, $t0, $t1
IF
ID
read
$t0, $t1
EX
MEM
WB
write
$s0
√

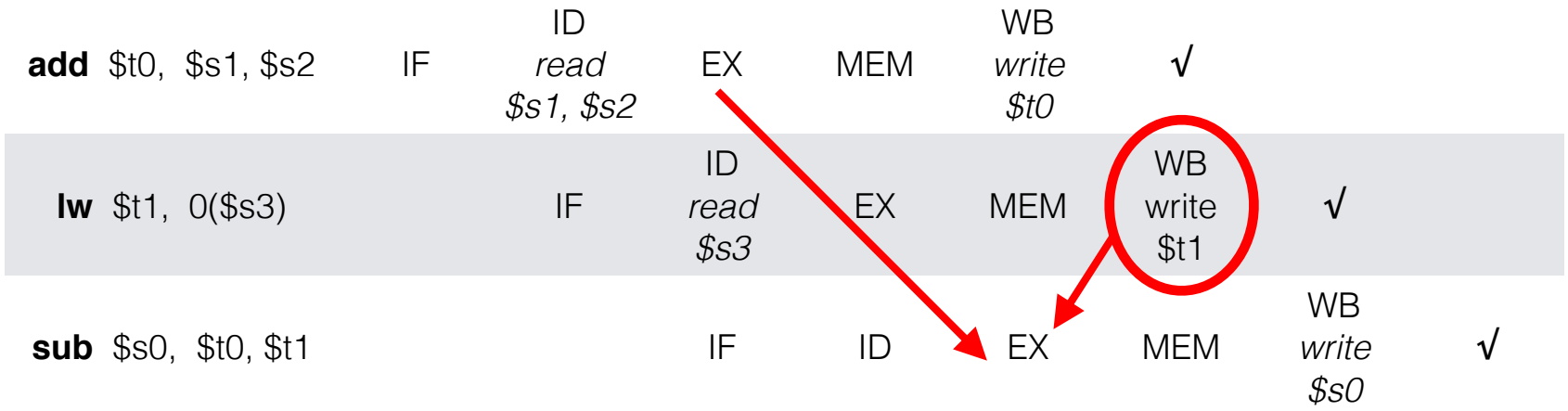
Solution: stall

add $t0, $s1, $s2	IF	ID *read $s1, $s2*	EX	MEM	WB *write $t0*	√			
add $t1, $s3, $s4		IF	ID *read $s3, $s4*	EX	MEM	WB write $t1	√		
sub $s0, $t0, $t1			IF	ID	***(stall)***	***(stall)***	*read $t0, $t1*	EX	

Better: *forwarding / bypassing*



Another data hazard

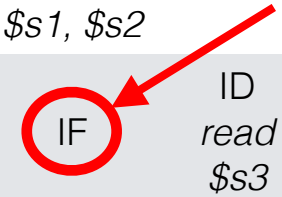


Data hazard with memory?

add $t0, $s1, $s2	IF	ID *read $s1, $s2*	EX	MEM	WB *write $t0*	√		
sw $t1, 0($s3)		IF	ID *read $s1, $t1*	EX	MEM	WB	√	
lw $s0, 0($s3)			IF	ID *read $s3*	EX	MEM	WB *write $s0*	√

Control hazard

beq $s1, $s2, *label*	IF	ID *read* *$s1, $s2*	EX	MEM	WB	√		
lw $t1, 0($s3)		IF	ID *read* *$s3*	EX	MEM	WB write $t1	√	
sub $s0, $t0, $t1			IF	ID	EX	MEM	WB *write* *$s0*	√



Mistakes harmless

beq $s1, $s2, *label*	IF	ID *read* *$s1, $s2*	EX	MEM	WB	√		
lw $t1, 0($s3)		IF	ID *read* *$s3*	EX	MEM	WB write $t1	√	
sub $s0, $t0, $t1			IF	ID	EX	MEM	WB *write* *$s0*	√

Non-uniform CPIs

IF	ID	EX	MEM	WB	√					
	IF	ID	**EX**	**EX**	**EX**	MEM	WB	√		
		IF	ID	***(WAIT)***	***(WAIT)***	EX	MEM	WB	√	
			IF	ID	***(WAIT)***	***(WAIT)***	EX	MEM	WB	√