

Figure 14.4a gives an example of this policy. We assume a superscalar pipeline capable of fetching and decoding two instructions at a time, having three separate functional units (e.g., two integer arithmetic and one floating-point arithmetic), and having two instances of the write-back pipeline stage. The example assumes the following constraints on a six-instruction code fragment:

- I1 requires two cycles to execute.
- I3 and I4 conflict for the same functional unit.
- I5 depends on the value produced by I4.
- I5 and I6 conflict for a functional unit.

Decode		Execute			Write		Cycle
I1	I2						1
I3	I4	I1	I2				2
I3	I4	I1					3
	I4			I3	I1	I2	4
I5	I6			I4			5
	I6		I5		I3	I4	6
			I6				7
					I5	I6	8

(a) In-order issue and in-order completion

Decode		Execute			Write		Cycle
I1	I2						1
I3	I4	I1	I2				2
	I4	I1		I3	I2		3
I5	I6			I4	I1	I3	4
	I6		I5		I4		5
			I6		I5		6
					I6		7

(b) In-order issue and out-of-order completion

Decode		Window	Execute		Write		Cycle	
I1	I2						1	
I3	I4	I1, I2	I1	I2			2	
I5	I6	I3, I4	I1		I3	I2	3	
		I4, I5, I6		I6	I4	I1	I3	4
		I5		I5		I4	I6	5
					I5		6	

(c) Out-of-order issue and out-of-order completion

Figure 14.4 Superscalar Instruction Issue and Completion Policies