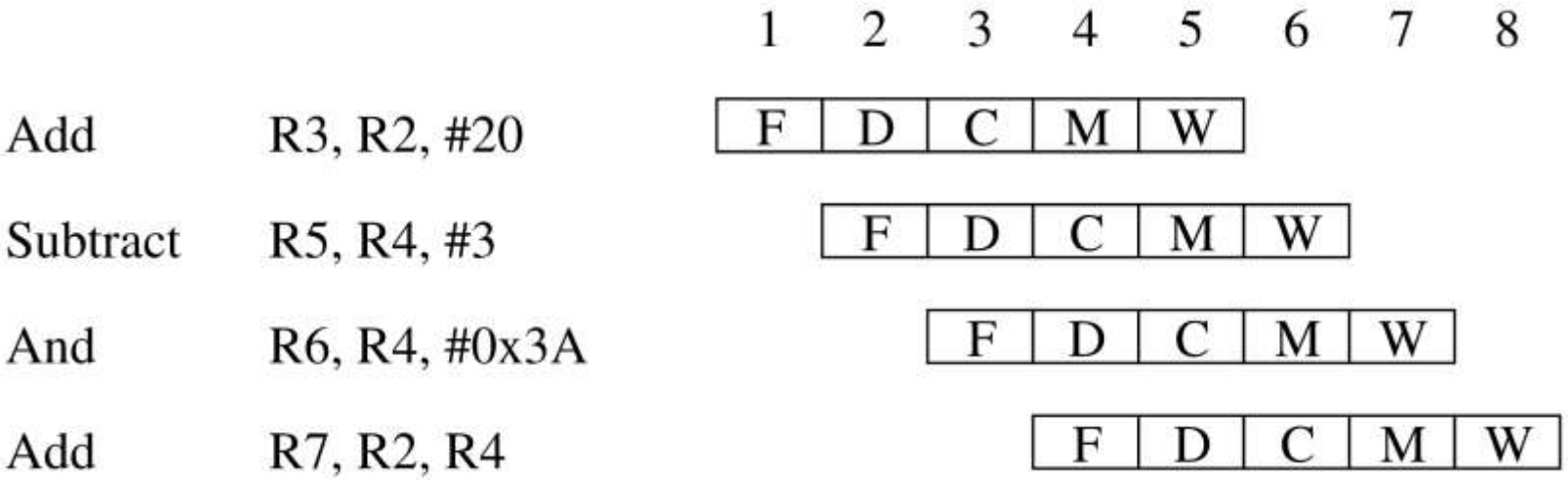
Chapter 6 6.1 Solution

(a) The pipeline execution diagram for the given code is shown below.



The description of activity in each stage during each cycle is given below. Activity for instructions prior to the first Add instruction and instructions after the second Add instruction is not described, but would nonetheless occur as determined by the type of instruction and the stage of the pipeline.

6.1 Solution

(a)

Cycle	Stage	Activity
1	F	fetching Add instruction
2	F	fetching Subtract instruction
	D	decoding Add instruction, reading register R2 (value 2000)
3	F	fetching And instruction
	D	decoding Subtract instruction, reading register R4 (value 50)
	C	performing arithmetic $2000 + 20 = 2020$ for Add instruction
4	F	fetching Add instruction
	D	decoding And instruction, reading register R4 (value 50)
	C	performing arithmetic $50 - 3 = 47$ for Subtract instruction
	M	no operation for Add instruction
5	D	decoding Add instruction, reading register R2 (value 200) and register R4 (value 50)
	C	performing logic operation $50 \wedge 3\mathrm{A}_{16} = 50$ for And instruction
	M	no operation for Subtract instruction
	W	write result of 2020 for Add instruction to register R3
6	C	performing arithmetic for Add instruction
	M	no operation for And instruction
	W	write result of 47 for Subtract instruction to register R5
7	M	no operation for Add instruction
	W	write result of 50 for And instruction to register R6
8	W	write result of 2050 for Add instruction to register R7

6.1 Solution

(*b*) The contents of each register during each cycle are described in the table below. The notation '–' is used to indicate that there is insufficient information to determine the register contents in that cycle. The instruction occupying the IR in each cycle is identified symbolically with an abbreviation.

注意：表格内容为周期一开始时寄存器的情况

RA,RB, RZ, RY寄存器内容都向前移一个周期。

	2	3	4	5	6	7	8
IR	Add	Sub	And	Add	–	–	–
PC	1004	1008	1012	1016	1020	1024	1028
RA	–	2000	50	50	2000	–	–
RB	–	–	–	–	50	–	–
RZ	–	–	2020	47	50	2050	–
RY	–	–	–	2020	47	50	2050

6.2 Solution

(a) The pipeline execution diagram for the given code is shown below.

```
                                1   2   3   4   5   6   7   8
Add     R3, R2, #20             F   D   C   M   W
Subtract R5, R4, #3                 F   D   C   M   W
And     R6, R3, #0x3A                   F   D   C   M   W
Add     R7, R2, R4                          F   D   C   M   W
```

Cycle	Stage	Activity
1	F	fetching Add instruction
2	F	fetching Subtract instruction
	D	decoding Add instruction, reading register R2 (value 2000)
3	F	fetching And instruction
	D	decoding Subtract instruction, reading register R4 (value 50)
	C	performing arithmetic $2000 + 20 = 2020$ for Add instruction

6.2 Solution

(a)

Cycle	Stage	Activity
4	F	fetching Add instruction
	D	decoding And instruction, *reading register R3 (value unknown)*
	C	performing arithmetic $50 - 3 = 47$ for Subtract instruction
	M	no operation for Add instruction
5	D	decoding Add instruction, reading register R2 (value 200) and register R4 (value 50)
	C	performing logic operation $2020 \wedge 3A = 32$ for And instruction
	M	no operation for Subtract instruction
	W	write result of 2020 for Add instruction to register R3
6	C	performing arithmetic for Add instruction
	M	no operation for And instruction
	W	write result of 47 for Subtract instruction to register R5
7	M	no operation for Add instruction
	W	write result of 32 for And instruction to register R6

6.2 Solution

(*b*) The contents of each register during each cycle are described in the table below.

注意：表格内容为周期一开始时寄存器的情况

RA,RB, RZ, RY寄存器内容都向前移一个周期。

	2	3	4	5	6	7	8
IR	Add	Sub	And	Add	–	–	–
PC	1004	1008	1012	1016	1020	1024	1028
RA	–	2000	*prev R3*	50	2000	–	–
RB	–	–	–	–	50	–	–
RZ	–	–	2020	47	50	2050	–
RY	–	–	–	2020	47	50	2050

32

32

The contents of RZ in cycle 4 and RY in cycle 5 are determined as follows:

$2020 \wedge 3A_{16} = 7E4_{16} \wedge 3A_{16} = 20_{16} = 32$

流水线部分补充题

Consider the following sequence of instructions being processed on the pipelined 5-stage RISC processor:

```
Load      R4, #100(R2)
Add       R5, R2, R3
Subtract  R6, R4, R5
And       R7, R2, R5
```

(1)Identify all the data dependencies in the above instruction sequence. For each dependency, indicate the two instructions and the register that causes the dependency.

Solution: There are three data dependencies in this instruction sequence

Subtract instruction depends on Load instruction for register R4

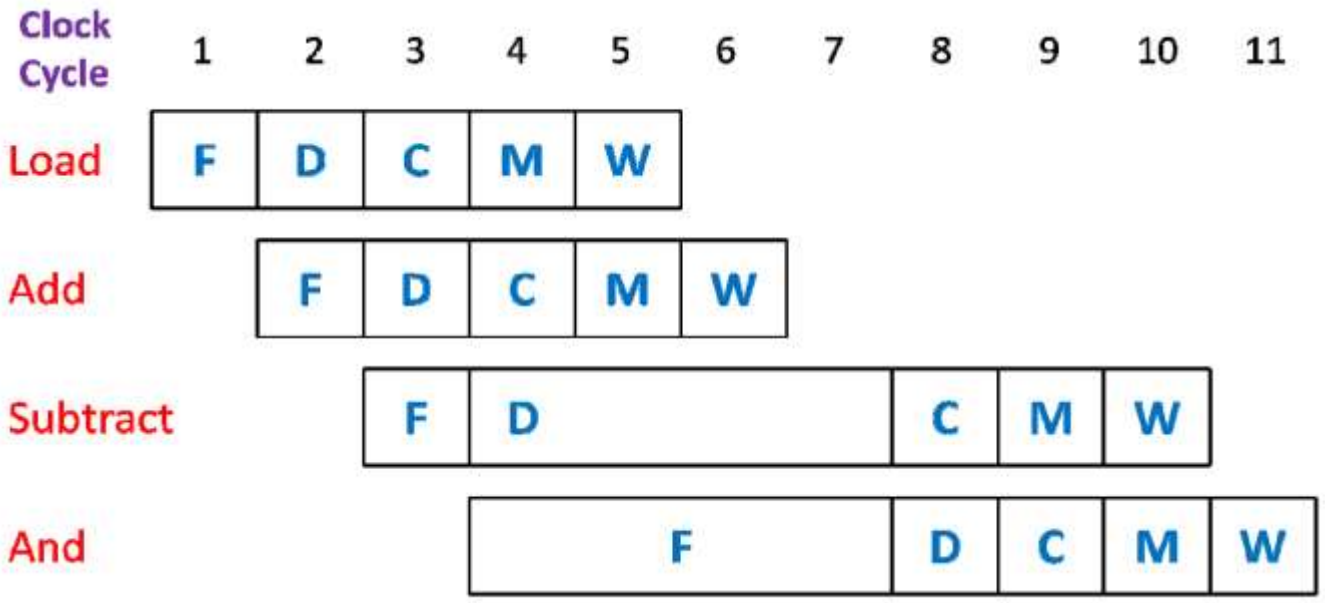
Subtract instruction depends on Add instruction for register R5

And instruction depends on Add instruction for register R5

补充题

(2)Assume that the pipeline does not use operand forwarding. Also assume that the only sources of pipeline stalls are the data hazards. Draw a diagram that represents instruction flow through the pipeline during each clock cycle.

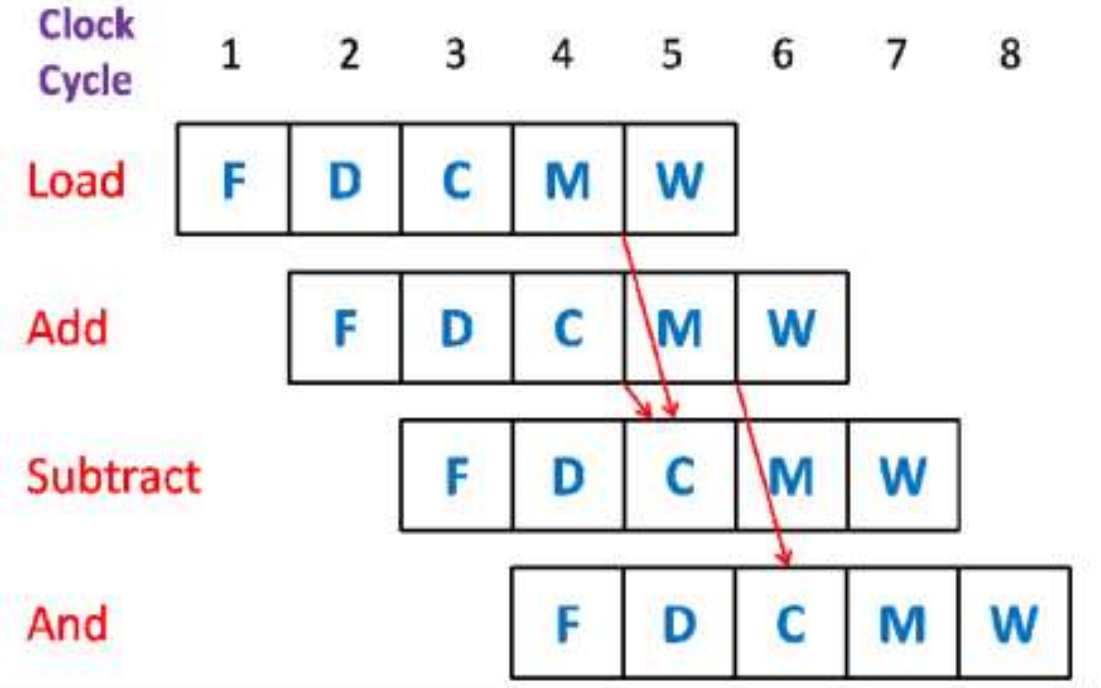
Solution: The following diagram shows the instruction flow through the pipeline.



补充题

(3)Assume that the pipeline uses operand forwarding. There are separate forwarding paths from the outputs of stage-3 and stage-4 to the input of stage-3. Draw a diagram that represents the flow of instructions through the pipeline during each clock cycle. Indicate operand forwarding by arrows.

Solution: The following diagram shows the instruction flow through the pipeline in the presence of operand forwarding:



3.1 Solution

To ensure that the given data are read only once.

3.3 Solution

A subroutine is called by a program instruction to perform a function needed by the calling program. An interrupt-service routine is initiated by an event such as an input operation or a hardware error. The function it performs may not be at all related to the program being executed at the time of interruption. Hence, it must not affect any of the data or status information relating to that program.

3.6 Solution

Setting the interrupt-enable bit in the PS last ensures that the processor will not be interrupted before it completes the initialization of all interrupts.

输入输出部分补充题

Three devices, A, B, and C, are connected to the bus of a computer. I/O transfers for all three devices use interrupt control. Interrupt nesting for devices A and B is not allowed, but interrupt requests from C may be accepted while either A or B is being serviced. Suggest different ways in which this can be accomplished in each of the following cases;

(a) the computer has one interrupt-request line

(b) two interrupt-request lines, INTR1 and INTR2, are available, with INTR1 having higher priority.

输入输出部分补充题

(b) two interrupt-request lines, INTR1 and INTR2, are available, with INTR1 having higher priority.

Solution:

(a) Interrupts should be enabled, except when C is being serviced. The nesting rules can be enforced by manipulating the interrupt-enable flags in the interfaces of A and B.

(b) A and B should be connected to INTR2, and C to INTR1.

When an interrupt is received from either A or B, interrupts from the other device will be automatically disabled until the request has been serviced. However, interrupt requests from C will always be accepted.