

3.8 有一条动态多功能流水线由 5 段组成(如图 3.35 所示),加法用 1、3、4、5 段,乘法用 1、2、5 段,第 2 段的时间为 $2\Delta t$,其余各段的时间均为 Δt ,而且流水线的输出可以直接返回输入端或暂存于相应的流水寄存器中。若在该流水线上计算 $\sum_{i=1}^4 (A_i \times B_i)$, 试计算其吞吐率、加速比和效率。

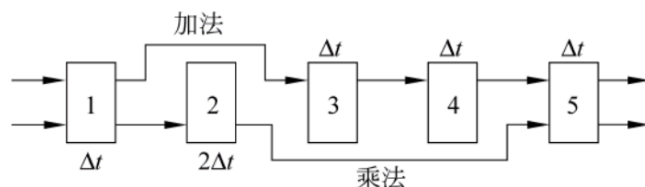
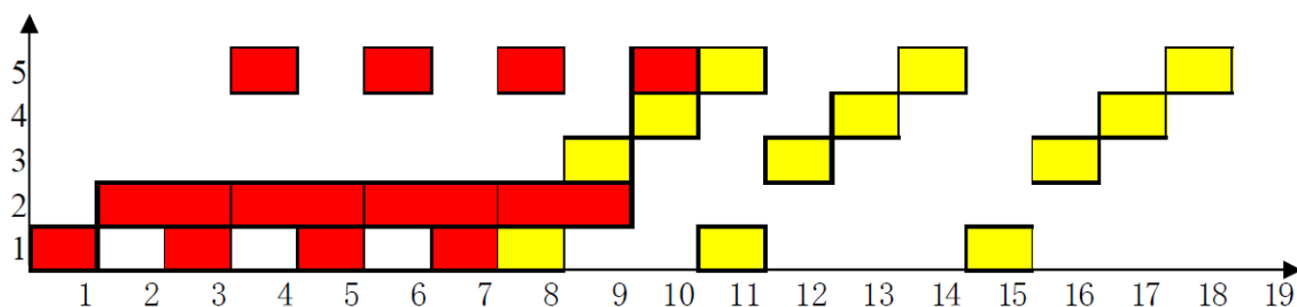


图 3.35 题 3.8 图

吞吐率, 加速比, 效率各10分

首先, 应选择适合于流水线工作的算法。对于本题, 应先计算 $A_1 \times B_1$ 、 $A_2 \times B_2$ 、 $A_3 \times B_3$ 和 $A_4 \times B_4$; 再计算 $(A_1 \times B_1) + (A_2 \times B_2)$ 和 $(A_3 \times B_3) + (A_4 \times B_4)$; 然后求总的结果。



由图可见, 它在 $18\Delta t$ 时间中, 给出了 7 个结果。所以吞吐率为:

$$TP = \frac{7}{18\Delta t}$$

如果不用流水线, 由于一次求积需 $3\Delta t$, 一次求和需 $4\Delta t$, 则产生上述 7 个结果共需 $(4 \times 4 + 3 \times 4) \Delta t = 28\Delta t$ 。所以加速比为:

$$S = \frac{28\Delta t}{18\Delta t} \approx 1.56$$

该流水线的效率可由阴影区的面积和 5 个段总时空区的面积的比值求得:

$$E = \frac{4 \times 4 + 3 \times 4}{5 \times 18} \approx 0.31$$

3.10 有一个 5 段流水线,各段执行时间均为 Δt ,其预约表如表 3.9 所示。

表 3.9 题 3.10 表格

时间 功能段	1	2	3	4	5	6	7
S_1	✓						✓
S_2		✓			✓		
S_3			✓	✓			
S_4				✓			✓
S_5					✓	✓	

(1) 画出流水线任务调度的状态转移图。

(2) 分别求出允许不等时间间隔调度和等时间间隔调度的两种最优调度策略,计算这两种调度策略的流水线最大吞吐率。

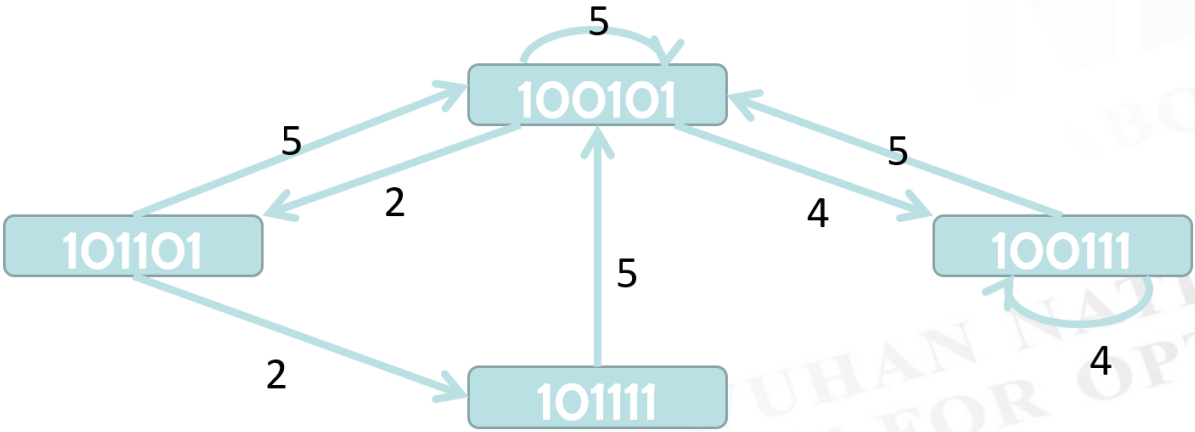
(3) 若连续输入 10 个任务,分别求采用这两种调度策略的流水线的实际吞吐率和加速比?

10分

(1) 由预约表可得禁止表: $F = \{6, 3, 1\}$;

根据禁止表得到初始冲突向量: $C0 = (100101)$;

流水线任务调度的状态转移图为:



10分

(2) 由状态转移图可得不发生段争用冲突的调度策略以及平均延迟时间如下所示。

调度策略	平均延迟时间	调度策略	平均延迟时间
(2,2,5)	$3\Delta t$	(4,5)	$4.5\Delta t$
(2,5)	$3.5\Delta t$	(5)	$5\Delta t$
(4)	$4\Delta t$		

由上可知，允许不等时间间隔调度的最优调度策略是(2, 2, 5)，流水线最大吞吐率为： $1/3\Delta t$ 。

等时间间隔的调度的最优调度策略是(4)，流水线最大吞吐率为： $1/4\Delta t$ 。

10分

(3) 按调度策略(2, 2, 5)，连续输入10 个任务的流水线实际吞吐率与加速比分别为：

$$TP_1 = \frac{10}{(2+2+5+2+2+5+2+2+5+7)\Delta t} = \frac{10}{34\Delta t}$$

$$S_1 = \frac{10 \times 7\Delta t}{34\Delta t} \approx 2.06$$

按调度策略(4)，连续输入10 个任务的流水线实际吞吐率与加速比分别为：

$$TP_2 = \frac{10}{(4 \times 9 + 7)\Delta t} = \frac{10}{43\Delta t}$$

$$S_2 = \frac{10 \times 7\Delta t}{43\Delta t} \approx 1.63$$

3.11 在 MIPS 流水线上运行以下代码序列。

```

LOOP: LW      R1,0(R2)
      DADDIU   R1,R1,#1
      SW       R1,0(R2)
      DADDIU   R2,R2,#4
      DSUB     R4,R3,R2
      BNEZ     R4,LOOP
    
```

其中,R3 的初值是 $R2+396$ 。假设：在整个代码序列的执行过程中,所有的存储器访问都是命中的,并且在一个时钟周期中对同一个寄存器的写操作和读操作可以通过分别把它们

安排在前半个时钟周期和后半个时钟周期来实现。请问：

(1) 在没有任何其他定向(或旁路)硬件的支持下,请画出该指令序列执行的流水线时空图。假设采用排空流水线的策略处理分支指令,且所有的存储器访问都命中 Cache,那么执行上述循环需要多少个时钟周期?

(2) 假设该流水线有正常的定向路径,请画出该指令序列执行的流水线时空图。假设采用预测分支失败的策略处理分支指令,且所有的存储器访问都命中 Cache,那么执行上述循环需要多少个时钟周期?

(3) 假设该流水线有正常的定向路径和一个单周期延迟分支,请对该循环中的指令进行调度,你可以重新组织指令的顺序,也可以修改指令的操作数,但是注意不能增加指令的条数。请画出该指令序列执行的流水线时空图,并计算执行上述循环所需要的时钟周期数。

10分

(1) 未改进方案：从每轮循环第1条指令开始到下轮循环第1条指令开始（此时最后一条指令**bnez r4,Loop**才执行完ID 周期）为15拍，见下图。末轮循环的最后一条指令（**bnez r4,Loop**）在ID周期后还执行3 拍才结束。

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
lw r1,0(r2)	IF	ID	EX	MEM	WB														
daddi r1,r1,1		IF	S	S	ID	EX	MEM	WB											
sw r1,0(r2)					IF	S	S	ID	EX	MEM	WB								
daddi r2,r2,4								IF	ID	EX	MEM	WB							
dsub r4,r3,r2									IF	S	S	ID	EX	MEM	WB				
bnez r4,Loop												IF	S	S	ID	EX	MEM	WB	★
halt															IF				
lw r1,0(r2)																IF	ID	EX	
15×99 + 3 = 1488																			

需要进行 $396/4=99$ 次循环，由于每次分支都清空流水线。从上图可以看出每次循环需要15个时钟周期，因此总共需要的时钟周期数= $15\times 99 + 3 = 1488$

或（在分支指令的MEM阶段后修改PC值）

寄存器读写可以定向，无其他旁路硬件支持。排空流水线。

指令	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
LW	IF	ID	EX	M	WB																	
DADDIU		IF	S	S	ID	EX	M	WB														
SW					IF	S	S	ID	EX	M	WB											
DADDIU								IF	ID	EX	M	WB										
DSUB									IF	S	S	ID	EX	M	WB							
BNEZ												IF	S	S	ID	EX	M	WB				
LW															IF	S	S	IF	ID	EX	M	WB

第*i*次迭代 ($i=0..98$) 开始周期: $1+(i\times 17)$

总的时钟周期数: $(98\times 17) + 18=1684$

10分

(2)采用定向技术，按“预测分支失败”的策略处理分支指令的。

从每轮循环第1条指令开始到下轮循环第1条指令开始为9拍，见下图。末轮循环的最后一条指令（bnez r4,Loop）在ID周期后还执行3拍才结束。

lw r1, 0(r2)	IF	ID	EX	MEM	WB													
daddi r1, r1, 1		IF	ID	S	EX	MEM	WB											
sw r1, 0(r2)			IF	S	ID	EX	MEM	WB										
daddi r2, r2, 4					IF	ID	EX	MEM	WB									
dsub r4, r3, r2						IF	ID	EX	MEM	WB								
bnez r4, Loop							IF	S	ID	EX	MEM	WB						★
halt									IF									
lw r1, 0(r2)												IF						
9 × 99 + 3 = 894																		

需要进行396/4=99次循环，由于每次分支都清空流水线。从上图可以看出每次循环需要9个时钟周期，因此总共需要的时钟周期数为 $9 \times 99 + 3 = 894$

或（在分支指令的MEM阶段后修改PC值）

有正常定向路径，预测分支失败。

指令	1	2	3	4	5	6	7	8	9	10	11	1	13	14	15
LW	IF	ID	EX	M	WB										
DADDIU		IF	ID	S	EX	M	WB								
SW			IF	S	ID	EX	M	WB							
DADDIU					IF	ID	EX	M	WB						
DSUB						IF	ID	EX	M	WB					
BNEZ							IF	ID	EX	M	WB				
LW								IF	miss	miss	IF	ID	EX	M	WB

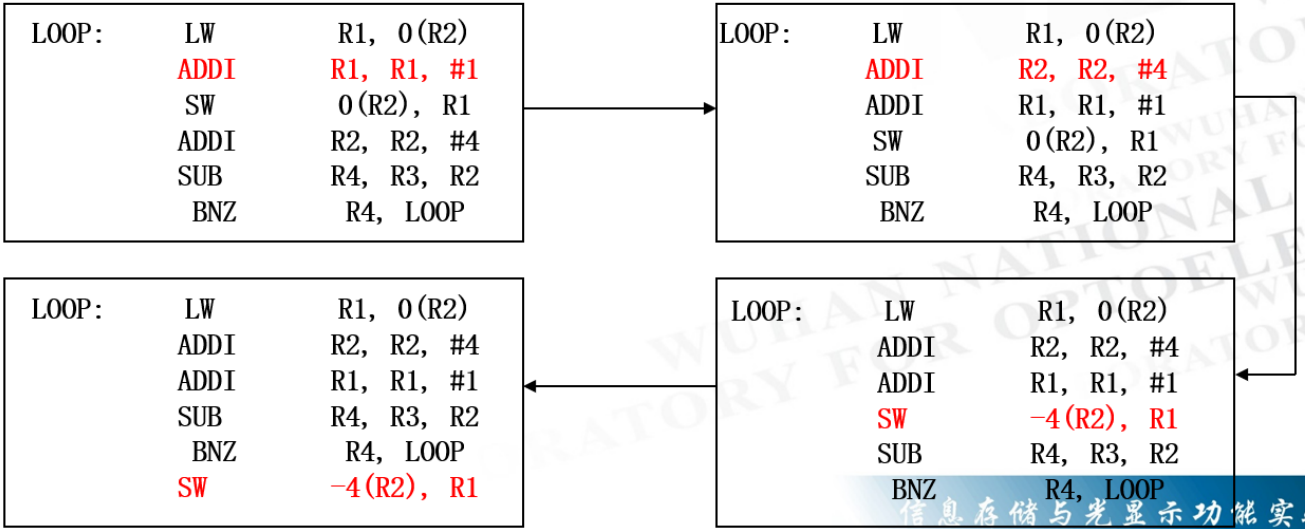
第 i 次迭代 (i=0..98) 开始周期: $1 + (i \times 10)$

总的时钟周期数: $(98 \times 10) + 11 = 991$

(3) 20分

指令执行重新排序如下：

lw	r1, 0(r2)	;加法寄存器R1←取数(R2)
addi	r2, r2, #4	;指针R2←指针R2+4
addi	r1, r1, #1	;R1←R1+1
Sub	r4, r3, r2	;R4←R3-R2
bnez	r4, Loop	;若R4≠0, 循环
sw	-4(r2), r1	;分支延迟槽, 存数(R2-4)←R1



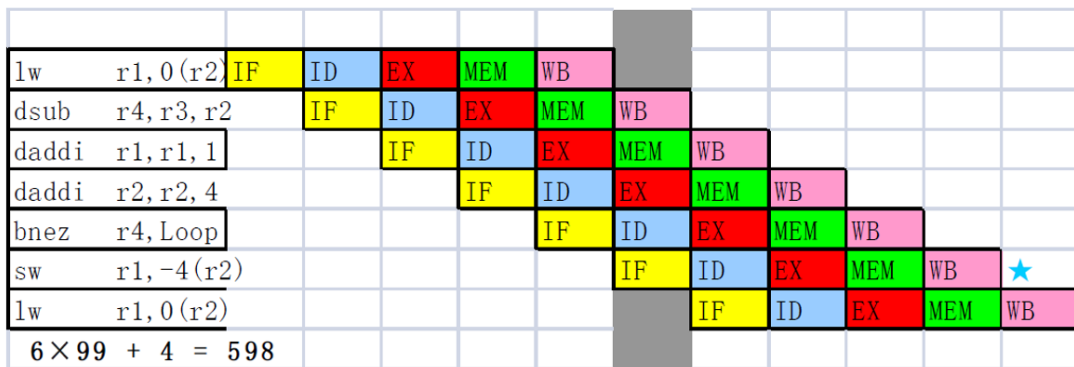
采用定向技术、单周期延迟分支、指令调度技术，重新安排指令顺序如下：

```

loop: lw  r1, 0(r2)
      addi r2, r2, #4
      addi r1, r1, #1
      sub  r4, r3, r2
      bnez r4, loop
      sw   r1, -4(r2)
    
```

从每轮循环第1 条指令开始到下轮循环第1 条指令开始为6 拍，见下图。末轮循环的最后一条指令（sw -4(r2), r1）在ID周期后还执行4 拍才结束。

所以，总拍数= 6×99 + 4 = 598



实验字