

## Tomasulo 算法

考虑如下的一段汇编程序：

```
L. D    F2, 0(R1)
L. D    F3, 0(R2)
MUL.D F4, F2, F3
ADD.D F3, F4, F3
ADD.D F5, F5, F2
DIV.D  F6, F5, F3
SUB.D  F3, F6, F2
```

其中 L.D 表示 LOAD 指令，它需要用 3 个时钟周期完成；MUL.D 表示乘法指令；它需要用 10 个时钟周期完成；ADD.D 表示加法指令，它需要用 2 个时钟周期完成；SUB.D 表示减法指令，它需要用 2 个时钟周期完成；DIV.D 表示除法指令，它需要用 40 个时钟周期完成。

已知保留站（reservation stations）包括：三个加减法保留站（Add1, Add2, Add3），两个乘除法保留站（Mult1, Mult2），三个 Load Buffer，此外共有 7 个浮点寄存器分别标记为 F0 到 F6。

根据 Tomasulo 算法：

**注意：**

1. 请同学们注意书写规范，尽可能和课件上保持一致，比如在每一个保留站的 OP 位，应当标注出指令的名字（类似 ADD.D，MUL.D）而非保留站的名字 Add，Mult 等。
2. 需要注意在第四条指令发射时，此时由于第二条 L.D 还没有执行结束，F3 中依旧为 Load 2，但是 ADD.D 要写 F3，应当用 Add 1 来覆盖 Load 2（想想看，为什么？答案在于寄存器状态是为了给对于发射指令存入保留站中的时候，如果需要以某一个寄存器为其操作数的来源，那么就将寄存器状态中的数值拷贝入保留站中。所以在这个例子中，如果 ADD.D 要写入 F3，未来的指令指令只要是引用 F3 其语义显然是要使用 ADD.D 的结果而不是之前的 L.D 的结果，因此这里必然需要使用 Add 1 来覆盖 Load 2）本题设计的初衷在于希望同学们通过这样的过程真正 Tomasulo 算法的思想，而不仅仅是简单的抄一遍课件。

1)请填写各指令发射，执行完成，写回结果的周期：

	Issue	Exec Complete	Write Result
L. D    F2, 0(R1)	1	4	5
L. D    F3, 0(R2)	2	5	6
MUL.D F4, F2, F3	3	16	17
ADD.D F3, F4, F3	4	19	20
ADD.D F5, F5, F2	5	7	8
DIV.D  F6, F5, F3	6	60	61
SUB.D  F3, F6, F2	7	63	64

2) 请分别填写各个指令发射的那个周期，各个保留站和寄存器结果状态表格

**L. D F2, 0(R1)在第 1 周期发射**

	Busy	Address
Load 1	Yes	0+R1
Load 2	No	
Load 3	No	

	Busy	Op	S1 Vj	S2 Vk	RS Qj	RS Qk
Add 1	No					
Add 2	No					
Add 3	No					
Mult 1	No					
Mult 2	No					

	F0	F1	F2	F3	F4	F5	F6
FU			Load 1				

**L. D F3, 0(R2)在第 2 周期发射**

	Busy	Address
Load 1	Yes	0+R1
Load 2	Yes	0+R2
Load 3	No	

	Busy	Op	S1 Vj	S2 Vk	RS Qj	RS Qk
Add 1	No					
Add 2	No					
Add 3	No					
Mult 1	No					
Mult 2	No					

	F0	F1	F2	F3	F4	F5	F6
FU			Load 1	Load 2			

**MUL.D F4, F2, F3 在第 3 周期发射**

	Busy	Address
Load 1	Yes	0+R1
Load 2	Yes	0+R2
Load 3		

	Busy	Op	S1 Vj	S2 Vk	RS Qj	RS Qk
Add 1	No					
Add 2	No					
Add 3	No					
Mult 1	Yes	MUL.D			Load 1	Load 2
Mult 2	No					

	F0	F1	F2	F3	F4	F5	F6
FU			Load 1	Load 1	Mult 1		

ADD.D F3, F4, F3 在第 4 周期发射

	Busy	Address
Load 1	Yes	0+R1
Load 2	Yes	0+R2
Load 3		

	Busy	Op	S1 Vj	S2 Vk	RS Qj	RS Qk
Add 1	Yes	ADD.D			Mult 1	Load 2
Add 2	No					
Add 3	No					
Mult 1	Yes	MUL.D			Load 1	Load 2
Mult 2	No					

	F0	F1	F2	F3	F4	F5	F6
FU			Load 1	Add 1	Mult 1		

ADD.D F5, F5, F2 在第 5 周期发射

	Busy	Address
Load 1	No	
Load 2	Yes	0 + R2
Load 3	No	

	Busy	Op	S1 Vj	S2 Vk	RS Qj	RS Qk
Add 1	Yes	ADD.D			Mult 1	Load 2
Add 2	Yes	ADD.D	R(F5)	M(A1)		
Add 3	No					
Mult 1	Yes	MULD	M(A1)			Load 2
Mult 2	No					

	F0	F1	F2	F3	F4	F5	F6
FU			M(A1)	Add 1	Mult 1	Add 2	

**DIV.D F6, F5, F3 在第 6 周期发射**

	Busy	Address
Load 1	No	
Load 2	No	
Load 3	No	

	Busy	Op	S1 Vj	S2 Vk	RS Qj	RS Qk
Add 1	Yes	ADD.D		M(A2)	Mult 1	
Add 2	Yes	ADD.D	R(F5)	M(A1)		
Add 3	No					
Mult 1	Yes	MUL.D	M(A1)	M(A2)		
Mult 2	Yes	DIV.D			Add 2	Add 1

	F0	F1	F2	F3	F4	F5	F6
FU			M(A1)	Add 1	Mult 1	Add 2	Mult 2

**SUB.D F3, F6, F2 在第 7 周期发射**

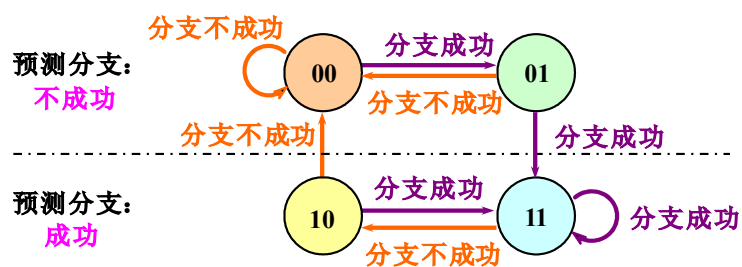
	Busy	Address
Load 1	No	
Load 2	No	
Load 3	No	

	Busy	Op	S1 Vj	S2 Vk	RS Qj	RS Qk
Add 1	Yes	ADD.D		M(A2)	Mult 1	
Add 2	Yes	ADD.D	R(F5)	M(A1)		
Add 3	Yes	SUB.D		M(A1)	Mult 2	
Mult 1	Yes	MUL.D	M(A1)	M(A2)		
Mult 2	Yes	DIV.D			Add 2	Add 1

	F0	F1	F2	F3	F4	F5	F6
FU			M(A1)	Add 3	Mult 1	Add 2	Mult 2

## 动态分支预测

假设 BHT 的状态转移图如下（引用自课件）：



已知如下指令序列

地址低两位	目标地址	是否跳转
01	b1	否
01	b1	否
01	b1	是
10	b2	否
10	b2	是

注意：

1. 请注意 BHT 并不是全局的，而是每一个分支指令都有一个自己的 BHT。本题中没有明确指出这一点确实是出题人的问题，在此向大家致歉，不过作业本身（至少在这门课中）的考察性就不强比如因为本题在作业中所扣的为数不多的一点分数基本不太可能直接反映到总评中，作业的主要目的在于提醒大家发现问题，提醒大家注意根本不存在全局只用一个 BHT 的情况，不然怎么能叫 BHT 呢？只需要叫 BH 不就可以了？

2. 有同学搞混了 BHT 表和 BTB 表的区别，BHT 表项只有两个二进制位，按照一个状态转移函数来实现状态转移，同目标跳转地址无关

设初始 BHT 状态为 00，求执行完每一步指令后的 BHT 状态。

指令标号（第几条指令后）	01 的 BHT	10 的 BHT
初始	00	00
1	00	00
2	00	00
3	01	00
4	01	00
5	01	01