1. un the following code on Scoreboard with function units including two FP multipliers，one FP divider, one FP adder that can perform float add and sub operations, and two integer units that is responsible for memory accessing, integer ALU operations and branch).

(1) Fill the following tables with the scoreboard status at the end of **5th** clock cycle when the first L.D instruction just finish WB stage. The scoreboard status includes instruction status, function units status and FP register status.

```
L.D      F0,0(R1)     ;  F0←MEM[R1+0]
L.D      F4,0(R2)     ;  F4←MEM[R2+0]
MUL.D    F6,F0,F4     ;  F6←F0 * F4
ADD.D    F8,F0,F2     ;  F8←F0 + F2
S.D      F6, 0(R3)    ;  MEM[R3 + 0] ← F6
S.D      F8, 0(R4)    ;  MEM[R4 + 0] ← F8
```

Instruction producing result	Instruction consuming result	Latency (in clock cycle)
FP operation	FP operation	3
FP operation	FP store（S.D）	2
FP load（L.D）	FP opereation	1
FP load（L.D）	FP store（S.D）	0

Instructions	发射（Issue）clock	读操作数(RO) clock	执行（EXE）clock	写结果(WB) clock
L.D F0, 0(R1)	1	2	3	5
L.D F4, 0(R2)	2	3	4	
MUL.D F6,F0,F4	3			
ADD.D F8,F0,F2	4			
S.D F6, 0(R3)				
S.D F8, 0(R4)				

Function units	Function Unit Status									
	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk	Address
Integer1	no	L.D	F0	R1				no		0+R1
Integer2	yes	L.D	F4	R2				no		0+R2
FPMult1	yes	MUL.D	F6	F0	F4			yes	no	
FPMult2										
FPDivider										
FPAdder	yes	ADD.D	F8	F0	F2			yes	yes	

	Register Status								
	F0	F2	F4	F6	F8	F10	F12	.....	F30
FU			Integer2	FPMult1	FPAdder				

Scoreboard是寄存器堆读写
所以一拍写下一拍读

(2) Fill the following table with instruction status at the end of **6th** Clock Cycle as the first line in the table.

	发射（Issue）	读操作数(RO)	执行(EXE)	写结果(WB)
L.D F0, 0(R1)	1	2	3	5
L.D F4, 0(R2)	2	3	4	6
MUL.D F6,F0,F4	3	~~6~~ CLK6 F4才WB		
ADD.D F8,F0,F2	4	6 CLK5	F0 WB (上一拍WB)	
S.D F6, 0(R3)	6			
S.D F8, 0(R4)				

WAW ✗

Store无写回

Function units	Function Unit Status									
	Busy	Op	Fi	Fj	Fk	Qj	Qk	Rj	Rk	A
Integer1	~~yes~~ no	S.D.		F6		MUL.D.		no		0+R3
Integer2	no									0+R2
FPMult1	yes	MUL.D	F6	F0	F4			~~yes~~ no	~~yes~~ no	
FPMult2										
FPDivider										
FPAdder	yes	ADD.D	F8	F0	F2			no	no	

	Register Status								
	F0	F2	F4	F6	F8	F10	F12	.....	F30
FU				FPMult1	FPAdder				

2. （30 points）For the following instruction sequence:

L.D F6, 34(R2)
L.D F2, 45(R3)
MUL.D F0, F2, F4
SUB.D F8, F2, F6
DIV.D F10, F0, F6
ADD.D F6, F8, F2

(1) Fill in the tables that Tomasulo algorithm used when the first instruction completes and finishes writing result. Assume the memory access unit needs two clock cycles to do execution: one for address calculation and one for memory access.

Time

Instruction status			
	ISSUE	EXECUTE	WRITE RESULT
1	1	3	4
2	2	4	
3	3		
4	4		
5			
6			

Reservation stations							
NAME	BUSY	Op	Vj	Vk	Qj	Qk	A
Load1	no	L.D.	[R2]				34+R2
Load2	yes	L.D.	[R3]				45+R3
Add1	yes	SUB.D		M[34+R2]	Load2		
Add2							
Add3							
Mult1	yes	MUL.D		[F4]	Load2		
Mult2							

Register status									
Field	F0	F2	F4	F6	F8	F10	F12	…	F30
Qi	Mult1	Load2			Add1				

L.D. F6, 34(R2)
L.D. F2, 45(R3)
MUL.D F0, F2, F4
SUB.D F8, F2, F6
DIV.D F10, F0, F6
ADD.D F6, F8, F2

Vj Vk得到WB后
下一拍开始EX

(2) Assume the following latencies: load is 1 clock cycle; add is 2 clock cycles; multiply is 10 clock cycles; divide is 40 clock cycles. Fill in the tables when the instruction MUL.D is about to write result (write result in next cycle).

Time

Instruction status			
	ISSUE Clock	EXECUTE finish Clock	WRITE RESULT Clock
1	1	3	4
2	2	4	5
3	3	6–10	
4	4	6–8	9
5	5		
6	6	10–12	13

Reservation stations							
NAME	BUSY	Op	Vj	Vk	Qj	Qk	A
Load1	no	L.D.	[R2]				34+R2
Load2	no	L.D.	[R3]				45+R3
Add1	no	SUB.D	M[45+R3]	M[34+R2]			
Add2	no	ADD.D	[F8]	M[45+R3]			
Add3							
Mult1	yes	MUL.D	M[45+R3]	[F4]			
Mult2	yes	DIV.D		M[34+R2]	Mult1		

Register status									
Field	F0	F2	F4	F6	F8	F10	F12	…	F30
Qi	Mult1					Mult2			

M[45+R3]