1. (28 pts) Show the state of the reservation stations and register file for each of the first 6 clock cycles. You may assume that add/sub and load/store take 1 cycle for execution and mult/div instructions take 15 cycles for execution. For the register file and the Q fields, you may identify instruction that is waited on with the instruction number. (HINT – remember that instructions begin execution on the clock cycle following the cycle that all values are available)

The state column represents the current execution state that the instruction is in. Possible states are: (I) Issue, (W) Waiting for operands, (R) Ready to execute (all operands are available), (E) Executing, and (C) Commit results. Note that an instruction may be in both Issued and Ready at the same time if all operands are available upon instruction issue. That instruction can go directly from Issued to Executing if other constraints are met.

The register file contains some starting values – r2 = 6, r3 = 8, r4 = 26. Please propagate all values through all subsequent clock cycles.

```
Inst #
A1      add     r2, r2, r3      # OP result, j, k (same format for all ALU instructions)
A2      load    r1, 0(r2)       # The value at memory location 0+r2 is 10
A3      mult    r3, r2, r1      # multiple r2 and r1 and store result in r3
A4      store   r1, 0(r2)
A5      sub     r5, r3, r4
```

Clock cycle 1:

Name	Inst. #	Vj	Vk	Qj	Qk	State
Add/Sub1	**A1**	**6**	**8**			**Issue**
Add/Sub2						
Add/Sub3						
Mult/Div1						
Mult/Div2						

Name	Inst #	Address	V (for stores)	Q (for stores)	State
Load/Store1					
Load/Store2					
Load/Store3					

r0	r1	r2	r3	r4	r5	r6
		A1	8	26		

Clock cycle 2:

Name	Inst. #	Vj	Vk	Qj	Qk	State
Add/Sub1	**A1**	**6**	**8**			**Execute**
Add/Sub2						
Add/Sub3						
Mult/Div1						
Mult/Div2						

Name	Inst #	Address	V (for stores)	Q (for stores)	State
Load/Store1	**A2**	**A1+0**			**Issue**
Load/Store2					
Load/Store3					

r0	r1	r2	r3	r4	r5	r6
	A2	**A1**	8	26		

Clock cycle 3:

Name	Inst. #	Vj	Vk	Qj	Qk	State
Add/Sub1	**A1**	**6**	**8**			**Commit**
Add/Sub2						
Add/Sub3						
Mult/Div1	**A3**	**14**			**A2**	**Issue**
Mult/Div2						

Name	Inst #	Address	V (for stores)	Q (for stores)	State
Load/Store1	**A2**	**14**			**Ready**
Load/Store2					
Load/Store3					

r0	r1	r2	r3	r4	r5	r6
	A2	**14**	**A3**	26		

Clock cycle 4:

Name	Inst. #	Vj	Vk	Qj	Qk	State
Add/Sub1						
Add/Sub2						
Add/Sub3						
Mult/Div1	**A3**	**14**			**A2**	**Wait**
Mult/Div2						

Name	Inst #	Address	V (for stores)	Q (for stores)	State
Load/Store1	**A2**	**14**			**Execute**
Load/Store2	**A4**	**14**		**A2**	**Issue**
Load/Store3					

r0	r1	r2	r3	r4	r5	r6
	A2	**14**	**A3**	26		

Clock cycle 5:

Name	Inst. #	Vj	Vk	Qj	Qk	State
Add/Sub1	**A5**		**26**	**A3**		**Issue**
Add/Sub2						
Add/Sub3						
Mult/Div1	**A3**	**14**	**10**			**Ready**
Mult/Div2						

Name	Inst #	Address	V (for stores)	Q (for stores)	State
Load/Store1	**A2**	**14**			**Commit**
Load/Store2	**A4**	**14**	**10**		**Ready**
Load/Store3					

r0	r1	r2	r3	r4	r5	r6
	10	**14**	**A3**	26	**A5**	

Clock cycle 6:

Name	Inst. #	Vj	Vk	Qj	Qk	State
Add/Sub1	**A5**		**26**	**A3**		**Wait**
Add/Sub2						
Add/Sub3						
Mult/Div1	**A3**	**14**	**10**			**Execute**
Mult/Div2						

Name	Inst #	Address	V (for stores)	Q (for stores)	State
Load/Store1					
Load/Store2	**A4**	**14**	**10**	**Execute**	
Load/Store3					

r0	r1	r2	r3	r4	r5	r6
	10	**14**	**A3**	26	**A5**	