The following program will be fed to the machine:

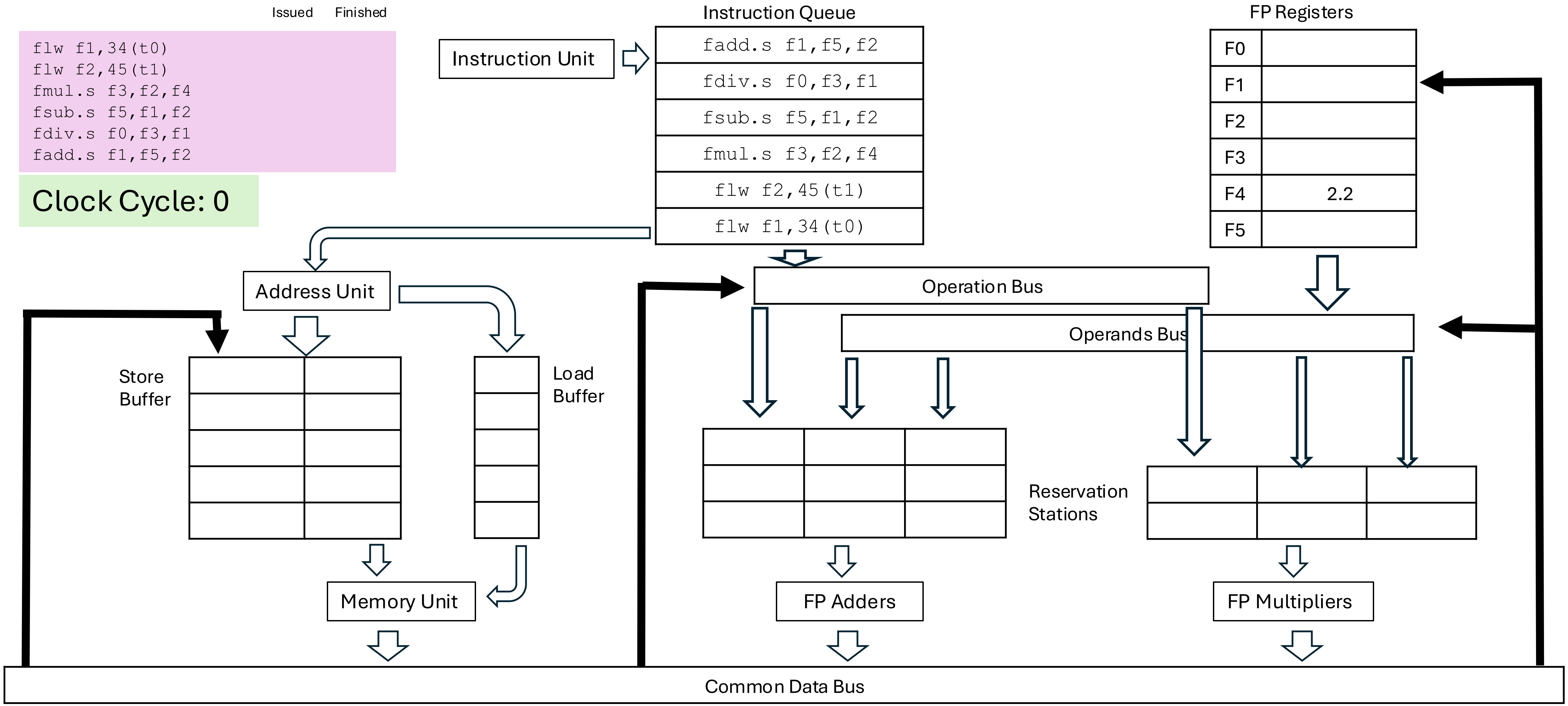
```
flw f1,34(t0)     # f1=load float from Memory[t0+34]
flw f2,45(t1)     # f2=load float from Memory[t1+45]
fmul.s f3,f2,f4   # f3=f2*f4
fsub.s f5,f1,f2   # f5=f1-f2
fdiv.s f0,f3,f1   # f0=f3/f1
fadd.s f1,f5,f2   # f1=f5+f2
```

- `flw`, `fsub` and `fadd` will take 2 clock cycles.
- `fmul` will take 10 clock cycles.
- `fdiv` instruction will take 30 clock cycles.

Issued Finished

```
flw f1,34(t0)
flw f2,45(t1)
fmul.s f3,f2,f4
fsub.s f5,f1,f2
fdiv.s f0,f3,f1
fadd.s f1,f5,f2
```

Clock Cycle: 0



Instruction Queue

```
fadd.s f1,f5,f2
fdiv.s f0,f3,f1
fsub.s f5,f1,f2
fmul.s f3,f2,f4
flw f2,45(t1)
flw f1,34(t0)
```

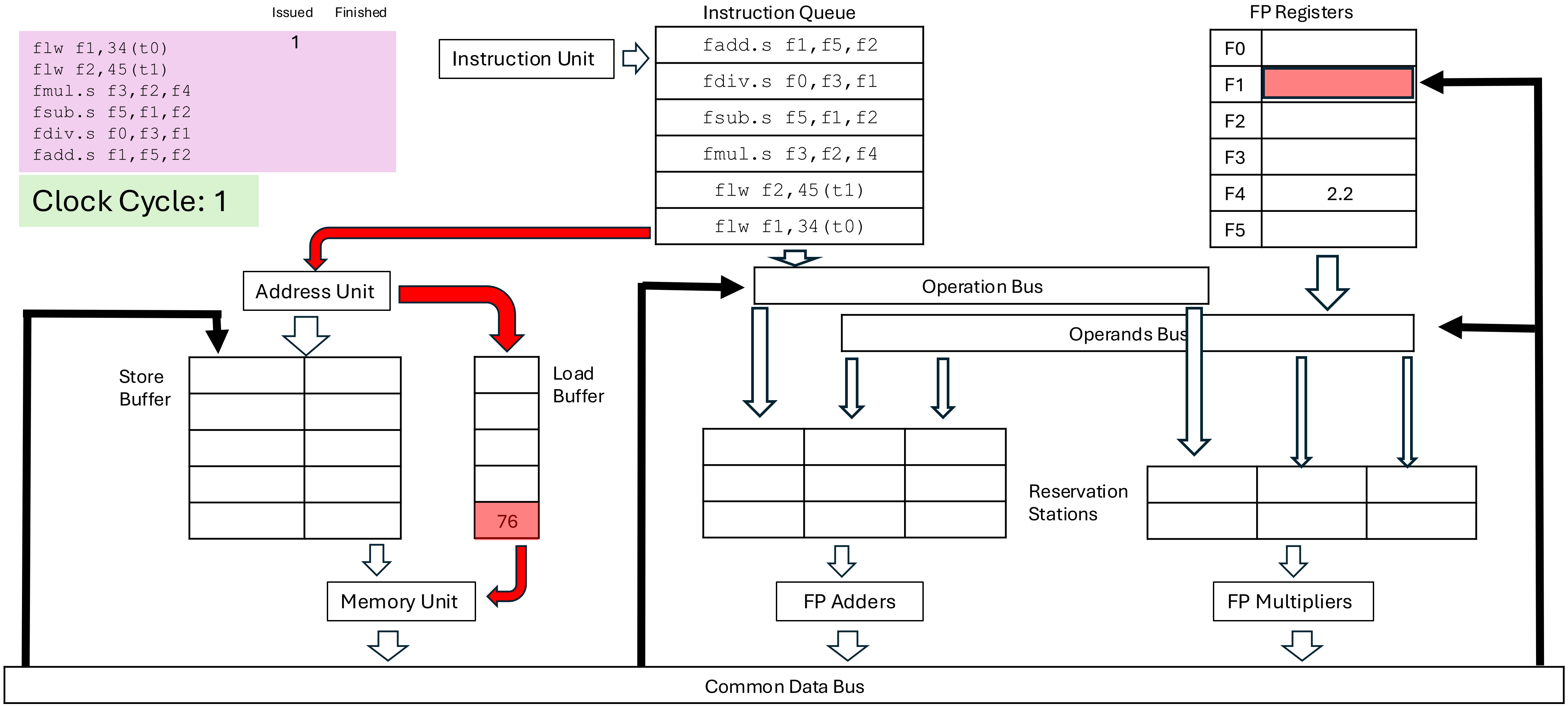
FP Registers

F0	
F1	
F2	
F3	
F4	2.2
F5	

At clock cycle 0, before we begin executing, things are as depicted. Integer registers t0=42 and t1=45. FP register F4 is 2.2.

	Issued	Finished
`flw f1,34(t0)`	1	
`flw f2,45(t1)`		
`fmul.s f3,f2,f4`		
`fsub.s f5,f1,f2`		
`fdiv.s f0,f3,f1`		
`fadd.s f1,f5,f2`		

Clock Cycle: 1



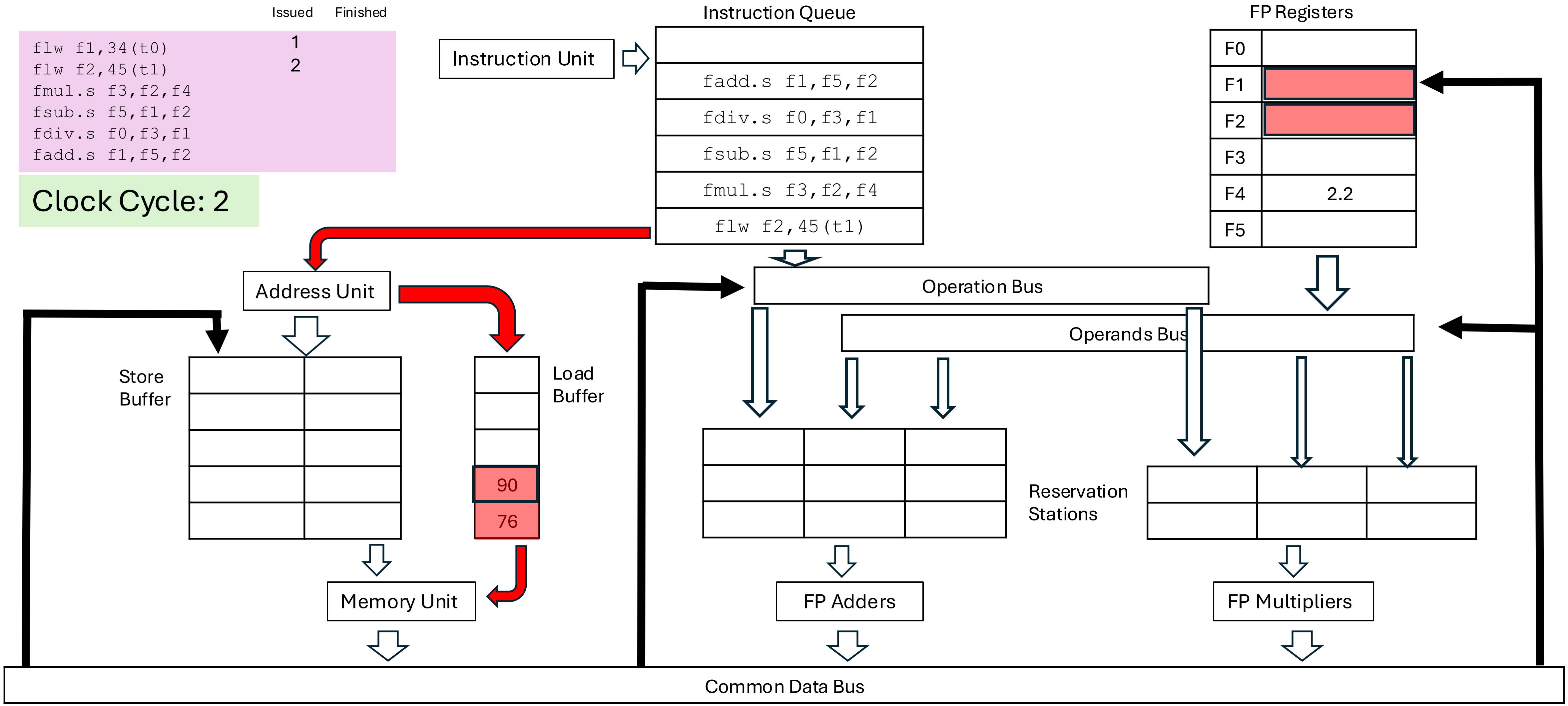
Instruction Queue
`fadd.s f1,f5,f2`
`fdiv.s f0,f3,f1`
`fsub.s f5,f1,f2`
`fmul.s f3,f2,f4`
`flw f2,45(t1)`
`flw f1,34(t0)`

FP Registers	
F0	
F1	🟥
F2	
F3	
F4	2.2
F5	

The first float load instruction is sent to the Address Unit where the address will be calculated (34+42=76) and sent to the first free load buffer, which is marked as busy (red). The destination of this operation (register F1) is also marked as busy.

	Issued	Finished
`flw f1,34(t0)`	1	
`flw f2,45(t1)`	2	
`fmul.s f3,f2,f4`		
`fsub.s f5,f1,f2`		
`fdiv.s f0,f3,f1`		
`fadd.s f1,f5,f2`		

Clock Cycle: 2

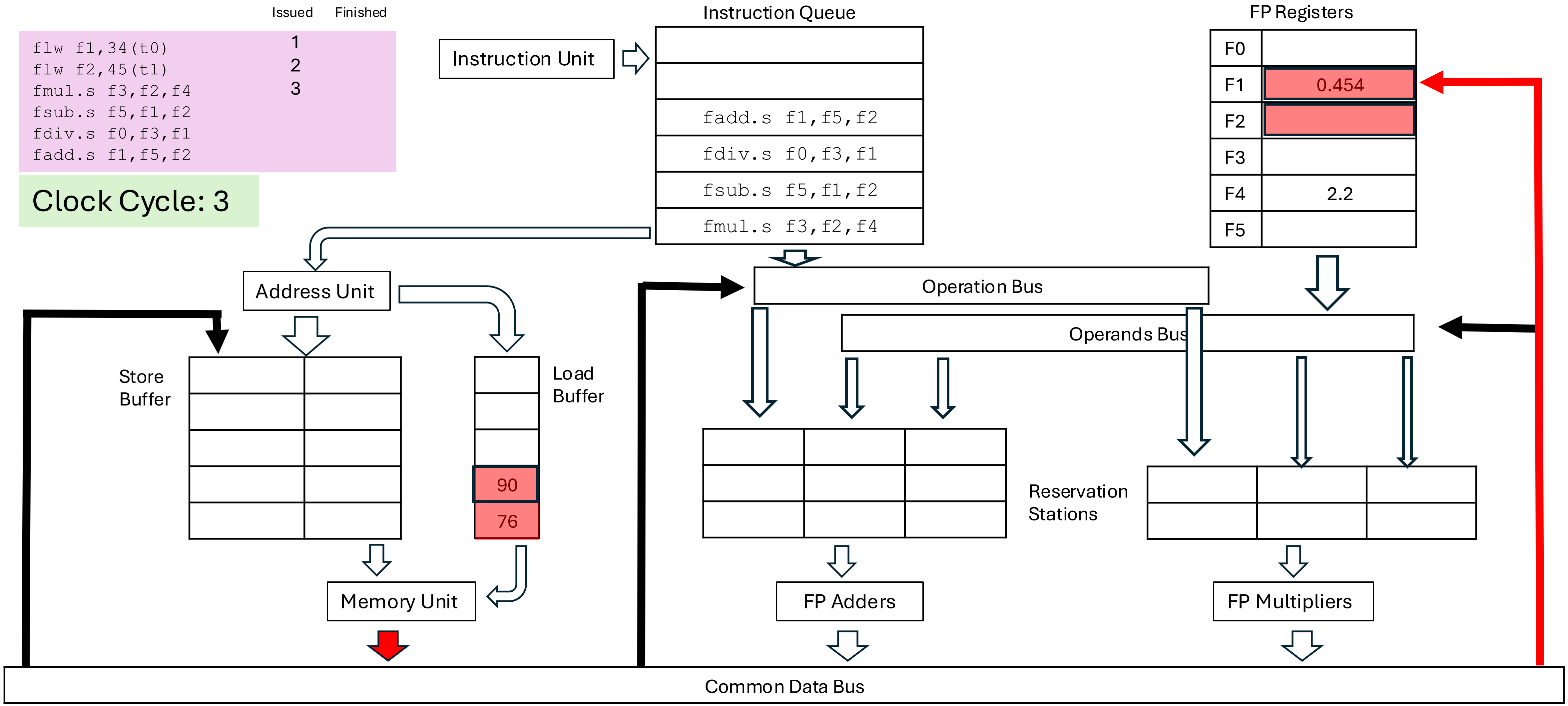


FP Registers	
F0	
F1	🟥
F2	🟥
F3	
F4	2.2
F5	

The second float load instruction (`flw`) is sent to the Address Unit where the address will be calculated (45+45=90) and sent to the first free load buffer, which is marked as busy (red). The destination of this operation (register F2) is also marked as busy.

	Issued	Finished
`flw f1,34(t0)`	1	
`flw f2,45(t1)`	2	
`fmul.s f3,f2,f4`	3	
`fsub.s f5,f1,f2`		
`fdiv.s f0,f3,f1`		
`fadd.s f1,f5,f2`		

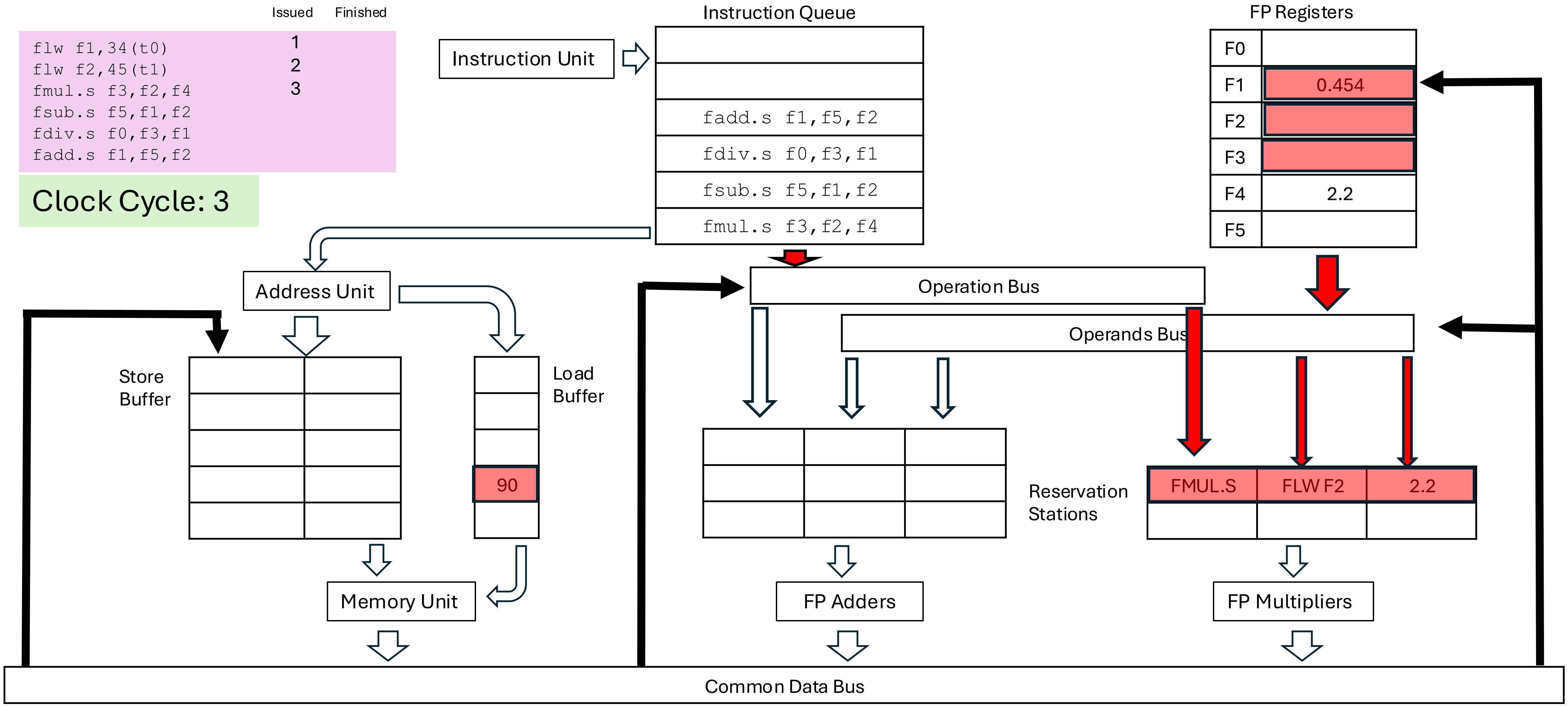
Clock Cycle: 3



The first float load instruction (`flw`) is ready and its result is broadcast to all subscribers. In this case, register F1.

	Issued	Finished
`flw f1,34(t0)`	1	
`flw f2,45(t1)`	2	
`fmul.s f3,f2,f4`	3	
`fsub.s f5,f1,f2`		
`fdiv.s f0,f3,f1`		
`fadd.s f1,f5,f2`		

Clock Cycle: 3



Instruction Queue

`fadd.s f1,f5,f2`
`fdiv.s f0,f3,f1`
`fsub.s f5,f1,f2`
`fmul.s f3,f2,f4`

FP Registers

F0	
F1	0.454
F2	
F3	
F4	2.2
F5	

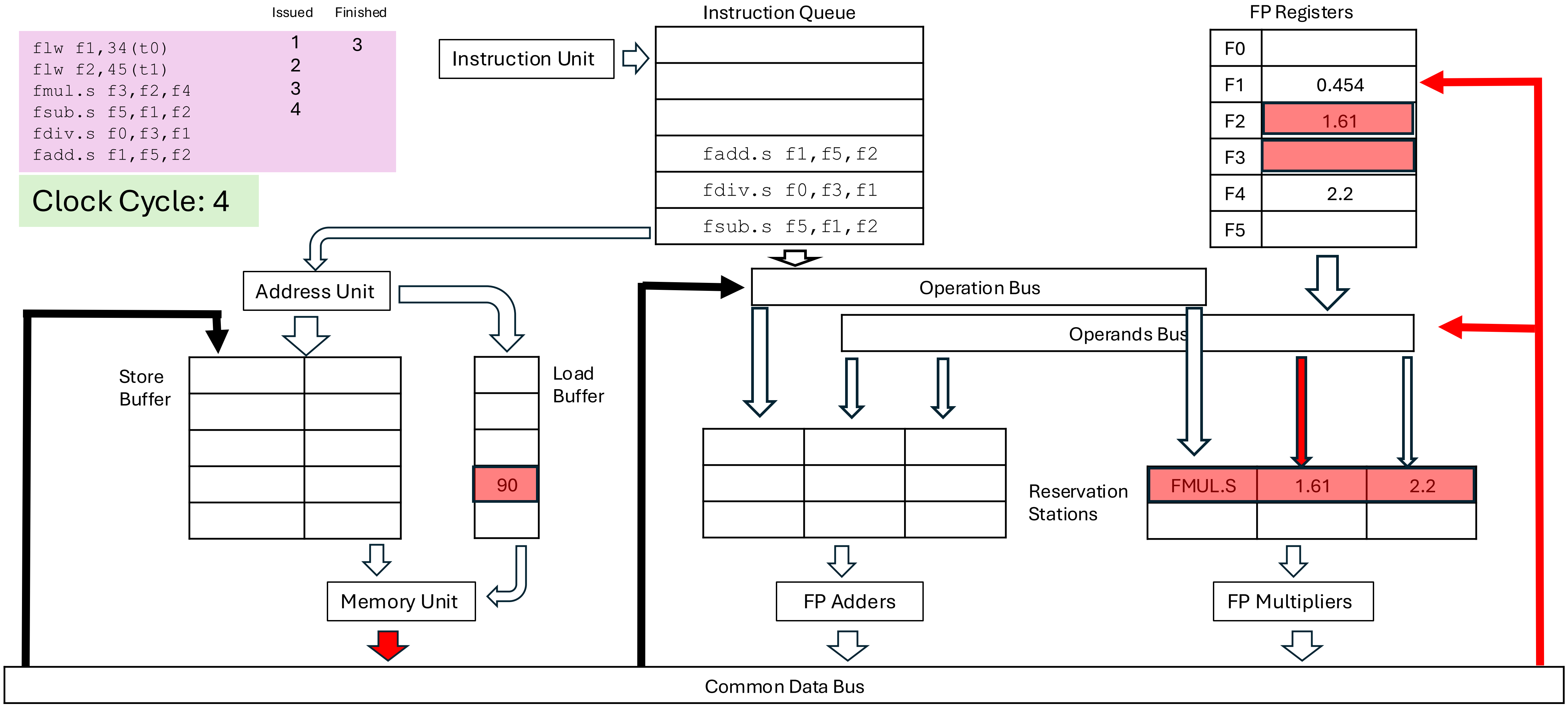
Reservation Stations (FP Multipliers)

FMUL.S	FLW F2	2.2

We are still in clock cycle 3. The `fmul.s` operation is ready to go through the pipeline to the first free multiply reservation station, but one of its operands (F2) isnot ready yet, so we mark the reservation station as busy and waiting for the `flw f2` operation. Destination register F3 is also marked as busy. In the load buffer, the completed load (`flw f1,34(t0)`) is marked as free.

	Issued	Finished
`flw f1,34(t0)`	1	3
`flw f2,45(t1)`	2	
`fmul.s f3,f2,f4`	3	
`fsub.s f5,f1,f2`	4	
`fdiv.s f0,f3,f1`		
`fadd.s f1,f5,f2`		

Clock Cycle: 4



Instruction Queue

```
fadd.s f1,f5,f2
fdiv.s f0,f3,f1
fsub.s f5,f1,f2
```

FP Registers

F0	
F1	0.454
F2	1.61
F3	
F4	2.2
F5	

Load Buffer: 90

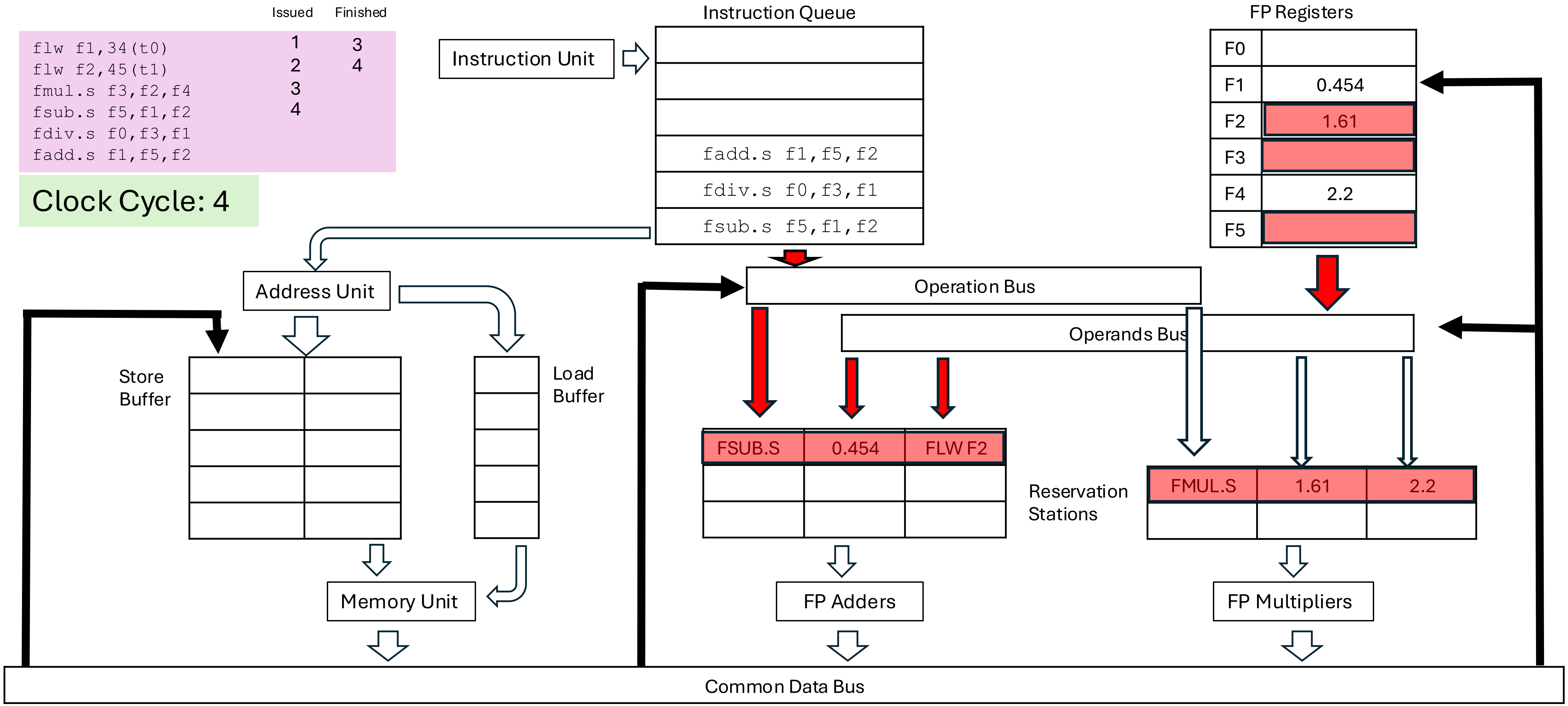
Reservation Stations

FMUL.S	1.61	2.2

We advance to clock cycle 4. Register F1 is no longer busy. The second load operation is read from the memory into the Common Data Bus and sent to its destination register F2, but also to the first busy multiply Reservation Station, since that Reservation Station is subscribed too.

	Issued	Finished
`flw f1,34(t0)`	1	3
`flw f2,45(t1)`	2	4
`fmul.s f3,f2,f4`	3	
`fsub.s f5,f1,f2`	4	
`fdiv.s f0,f3,f1`		
`fadd.s f1,f5,f2`		

Clock Cycle: 4



Instruction Queue

`fadd.s f1,f5,f2`
`fdiv.s f0,f3,f1`
`fsub.s f5,f1,f2`

FP Registers

F0	
F1	0.454
F2	1.61
F3	
F4	2.2
F5	

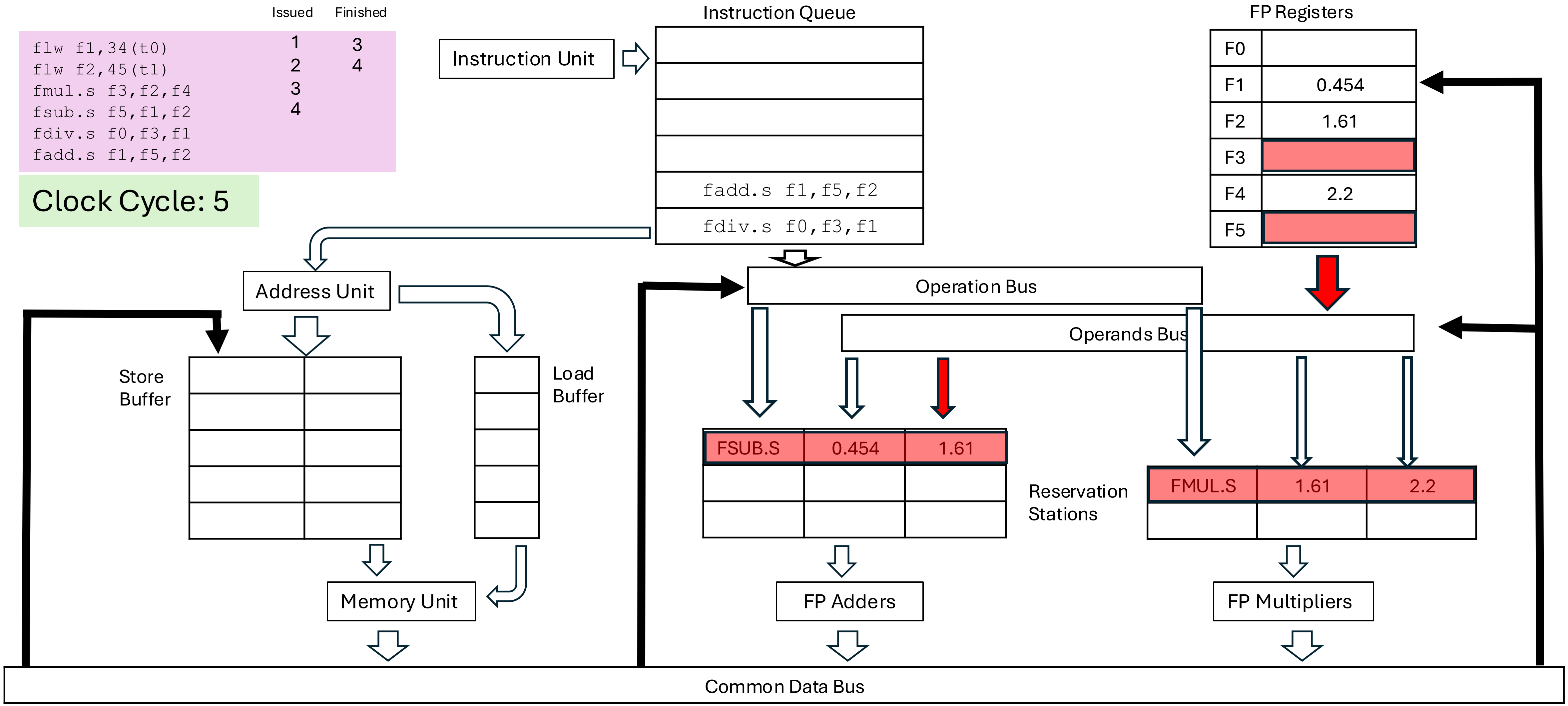
FSUB.S	0.454	FLW F2

FMUL.S	1.61	2.2

We are still in clock cycle 4. The `fsub` instruction continues down the pipeline. This substraction operation doesn't have all its operands since register F2 is still busy (the `flw f2` ends this cycle) and, we consider, it wasn't subscribed in time for the earlier Common Data Bus broadcast, so we mark that addition Reservation Station as busy and waiting for register F2 load operation. Destination register F5 is marked as busy. The Load Buffer is empty since the second load (`flw f2,45(t1)`) ended.

	Issued	Finished
`flw f1,34(t0)`	1	3
`flw f2,45(t1)`	2	4
`fmul.s f3,f2,f4`	3	
`fsub.s f5,f1,f2`	4	
`fdiv.s f0,f3,f1`		
`fadd.s f1,f5,f2`		

Clock Cycle: 5



FP Registers	
F0	
F1	0.454
F2	1.61
F3	🟥
F4	2.2
F5	🟥

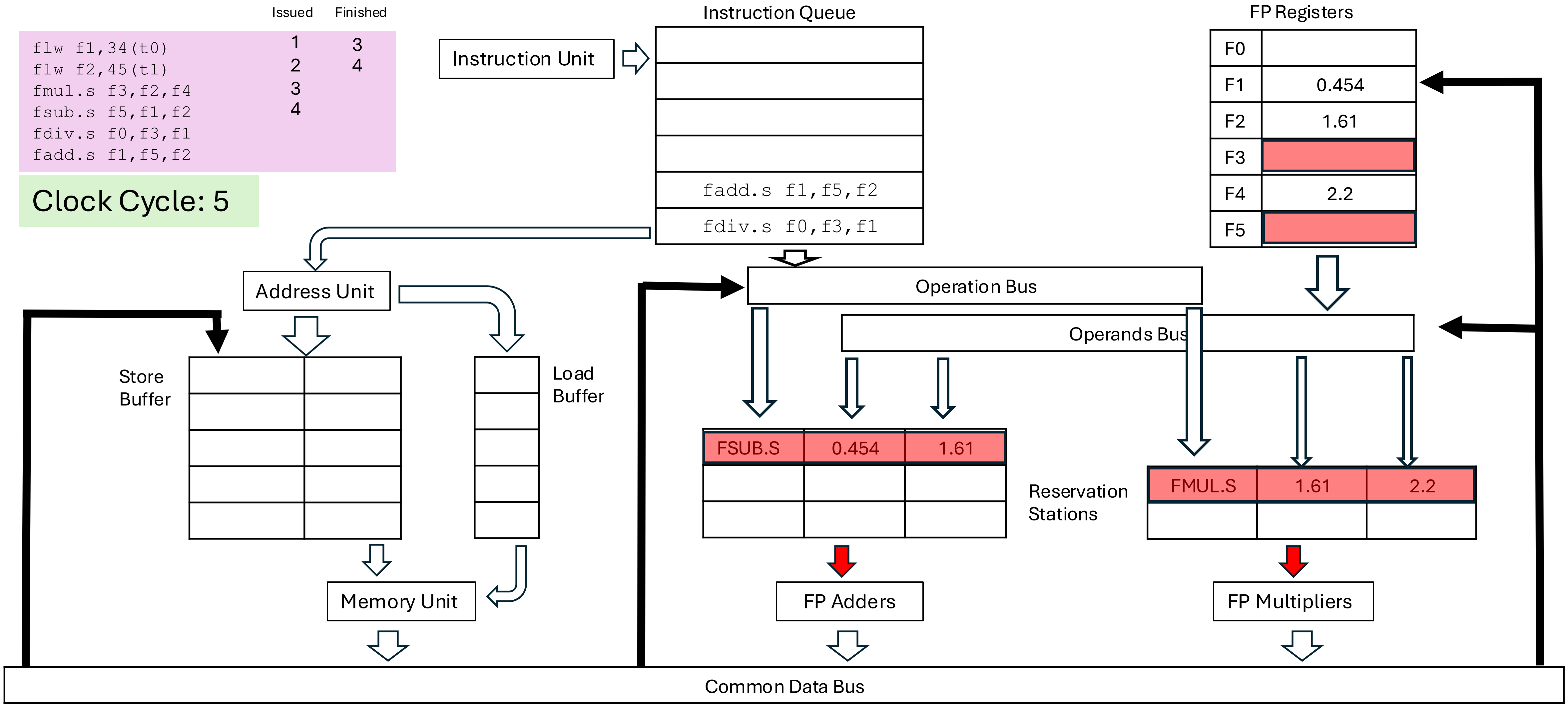
FSUB.S	0.454	1.61

FMUL.S	1.61	2.2

We in clock cycle 5. Register F2 is no longer busy, so its value is sent to the adding Reservation Station waiting for it.

	Issued	Finished
`flw f1,34(t0)`	1	3
`flw f2,45(t1)`	2	4
`fmul.s f3,f2,f4`	3	
`fsub.s f5,f1,f2`	4	
`fdiv.s f0,f3,f1`		
`fadd.s f1,f5,f2`		

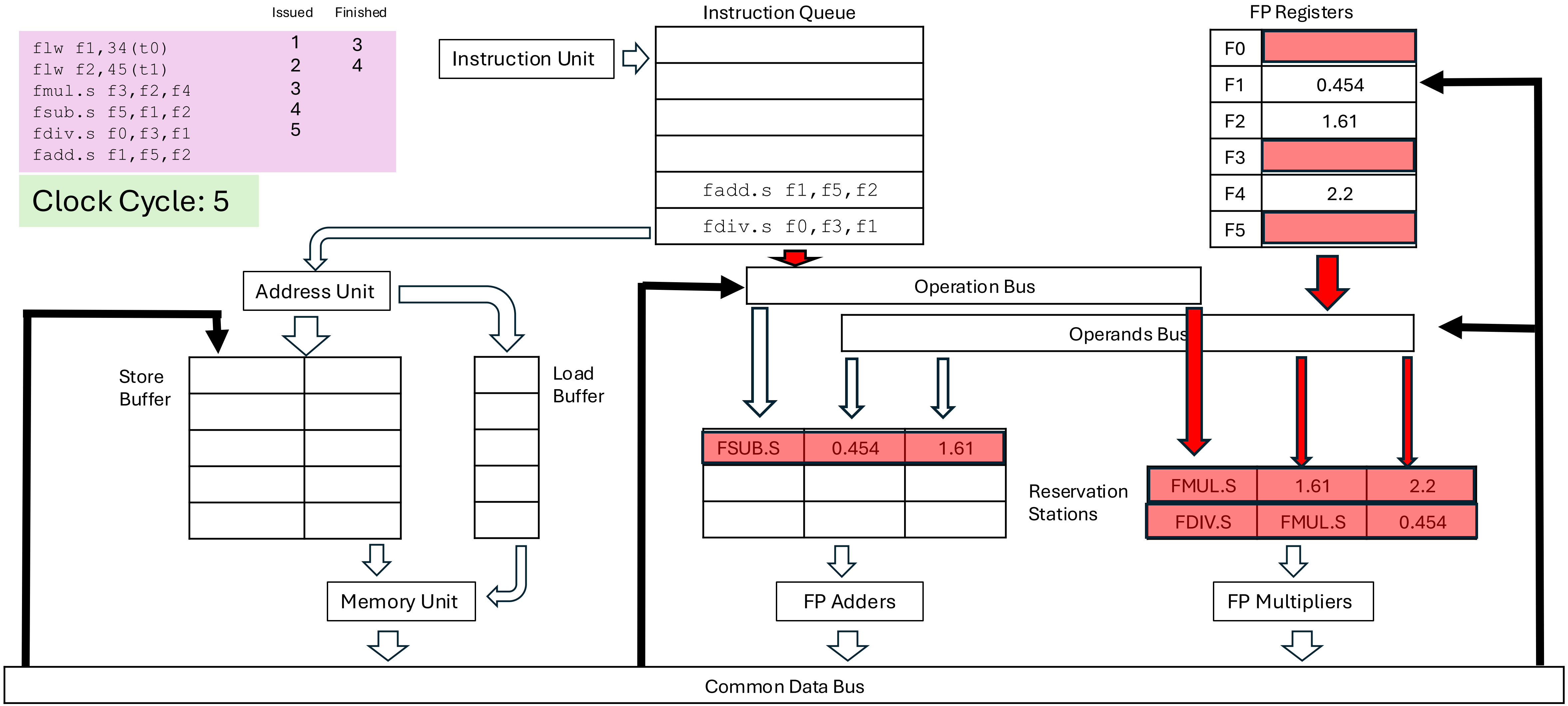
Clock Cycle: 5



Still in cycle 5, both the FSUB and FMUL in the reservation stations may begin executing. They are sent to the FP Adders/Multipliers.

	Issued	Finished
`flw f1,34(t0)`	1	3
`flw f2,45(t1)`	2	4
`fmul.s f3,f2,f4`	3	
`fsub.s f5,f1,f2`	4	
`fdiv.s f0,f3,f1`	5	
`fadd.s f1,f5,f2`		

Clock Cycle: 5



FP Registers

F0	🟥
F1	0.454
F2	1.61
F3	🟥
F4	2.2
F5	🟥

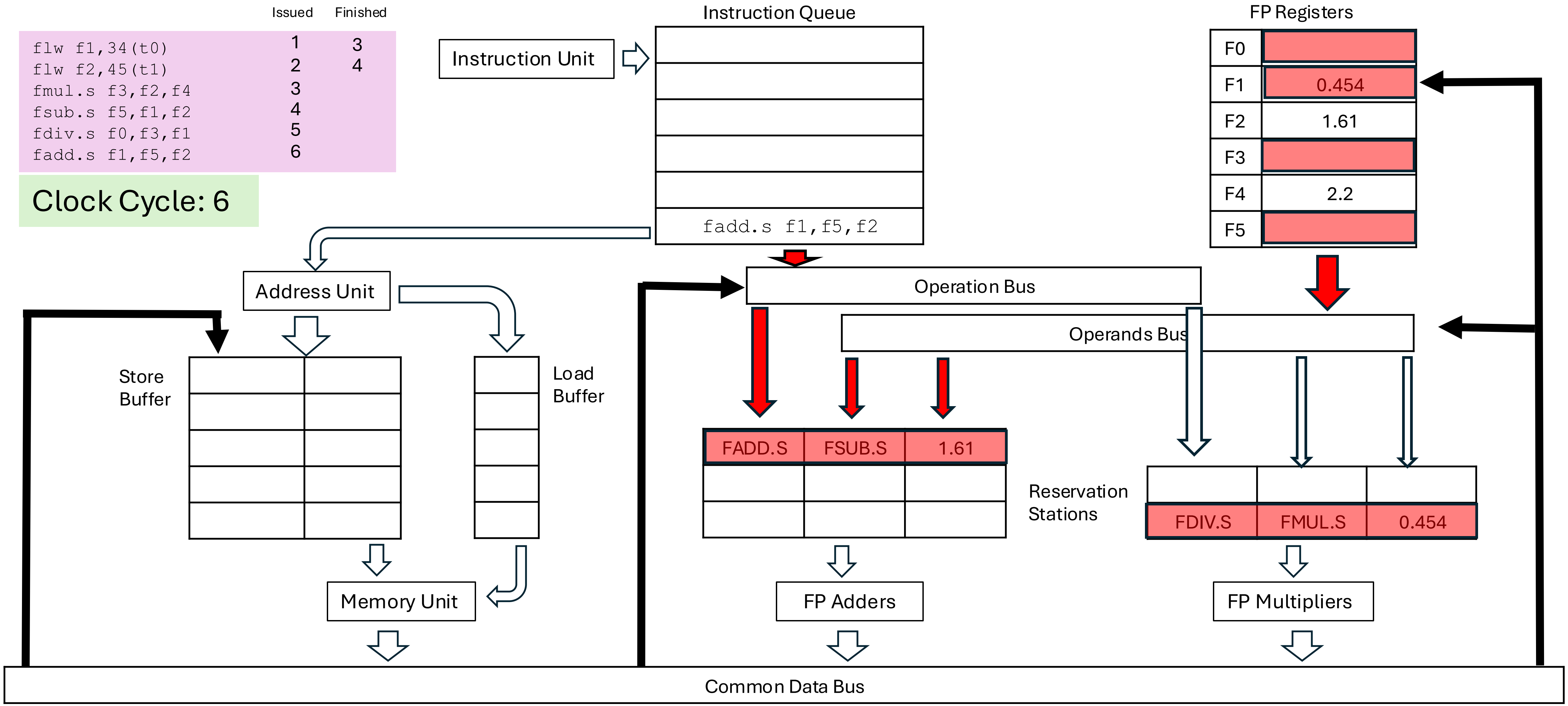
FSUB.S	0.454	1.61

FMUL.S	1.61	2.2
FDIV.S	FMUL.S	0.454

Still in cycle 5, the `fdiv.s` operation continues down the pipeline. We will need one more cycle to mark the occupied Reservation Stations as free, so we send this `fdiv` to the next free one. The `fdiv.s` must wait for the `fmul.s` operation, since it needs F3. We also need to mark its destination register, F0, as busy.

	Issued	Finished
`flw f1,34(t0)`	1	3
`flw f2,45(t1)`	2	4
`fmul.s f3,f2,f4`	3	
`fsub.s f5,f1,f2`	4	
`fdiv.s f0,f3,f1`	5	
`fadd.s f1,f5,f2`	6	

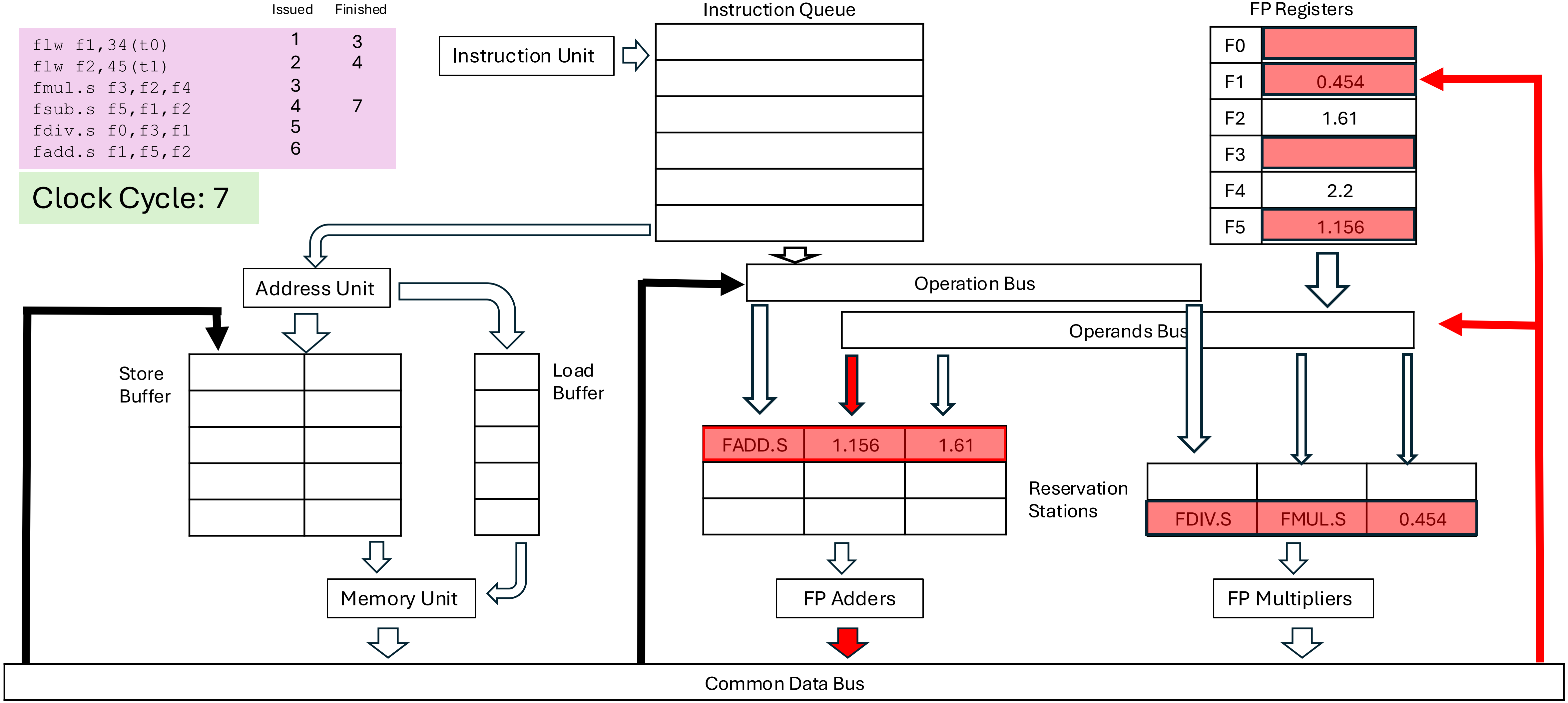
Clock Cycle: 6



In clock cycle 6 the Reservation Stations for FMUL and FSUB are free, since they were sent to execute. Registers F3, F5 and F0 are still busy waiting for FMUL, FSUB and FDIV. The `fadd f1` instruction continues down the pipeline to the first free adding Reservation Station. This instruction needs the results of the F5 register, that is waiting for the `fsub.s` operation to end, so we mark the Reservation Station as busy and waiting for `fsub.s`. We mark F1 as busy waiting for the `fadd.s` operation to complete.

	Issued	Finished
`flw f1,34(t0)`	1	3
`flw f2,45(t1)`	2	4
`fmul.s f3,f2,f4`	3	
`fsub.s f5,f1,f2`	4	7
`fdiv.s f0,f3,f1`	5	
`fadd.s f1,f5,f2`	6	

Clock Cycle: 7



FP Registers

F0	
F1	0.454
F2	1.61
F3	
F4	2.2
F5	1.156

Reservation Stations (FP Adders)

FADD.S	1.156	1.61

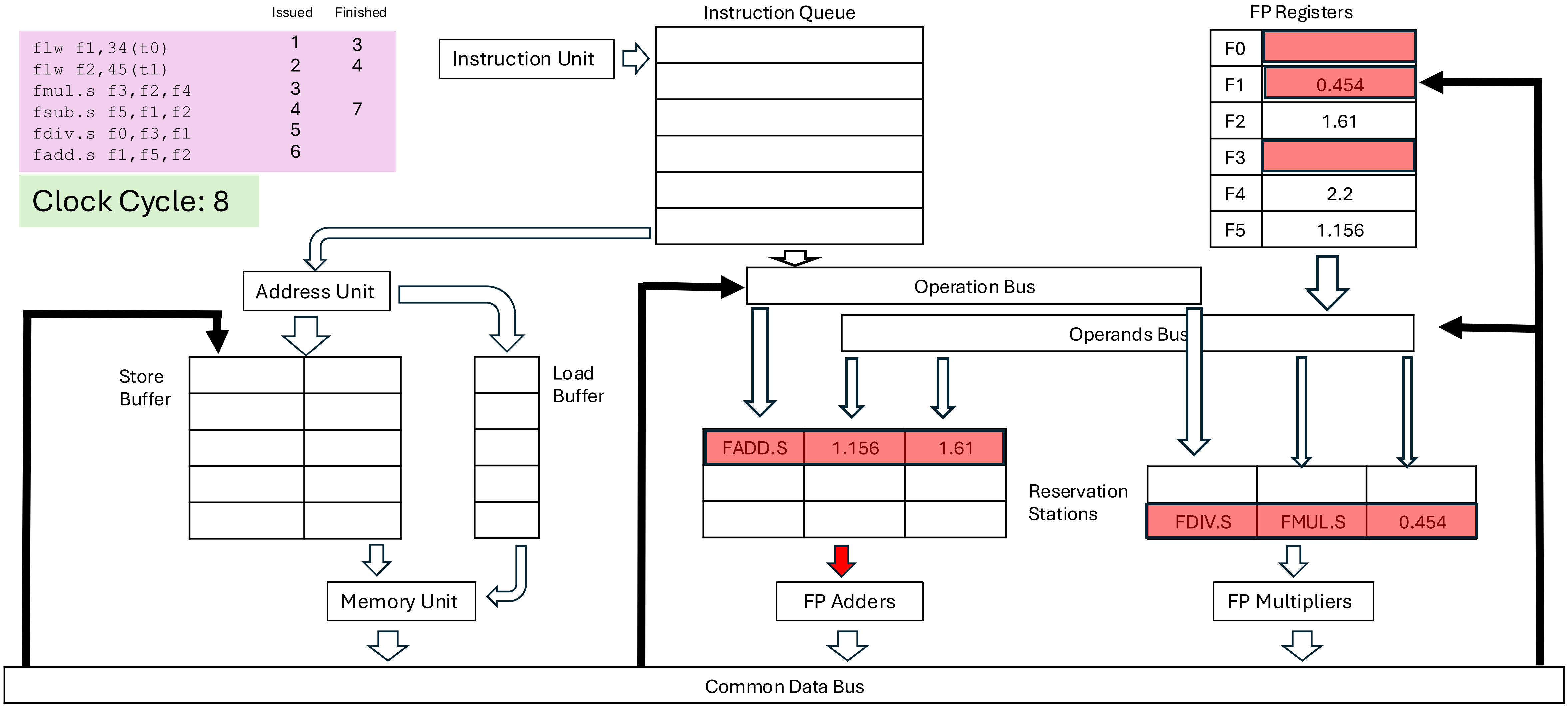
Reservation Stations (FP Multipliers)

FDIV.S	FMUL.S	0.454

In clock cycle 7 the `fsub.s f5,f1,f2` has ended (it takes 2 cycles), so its results are sent from the FP Adders to the Common Data Bus and all elements waiting forit, like the only FADD in the adding Reservation Stations and the F5 register.

	Issued	Finished
`flw f1,34(t0)`	1	3
`flw f2,45(t1)`	2	4
`fmul.s f3,f2,f4`	3	
`fsub.s f5,f1,f2`	4	7
`fdiv.s f0,f3,f1`	5	
`fadd.s f1,f5,f2`	6	

Clock Cycle: 8



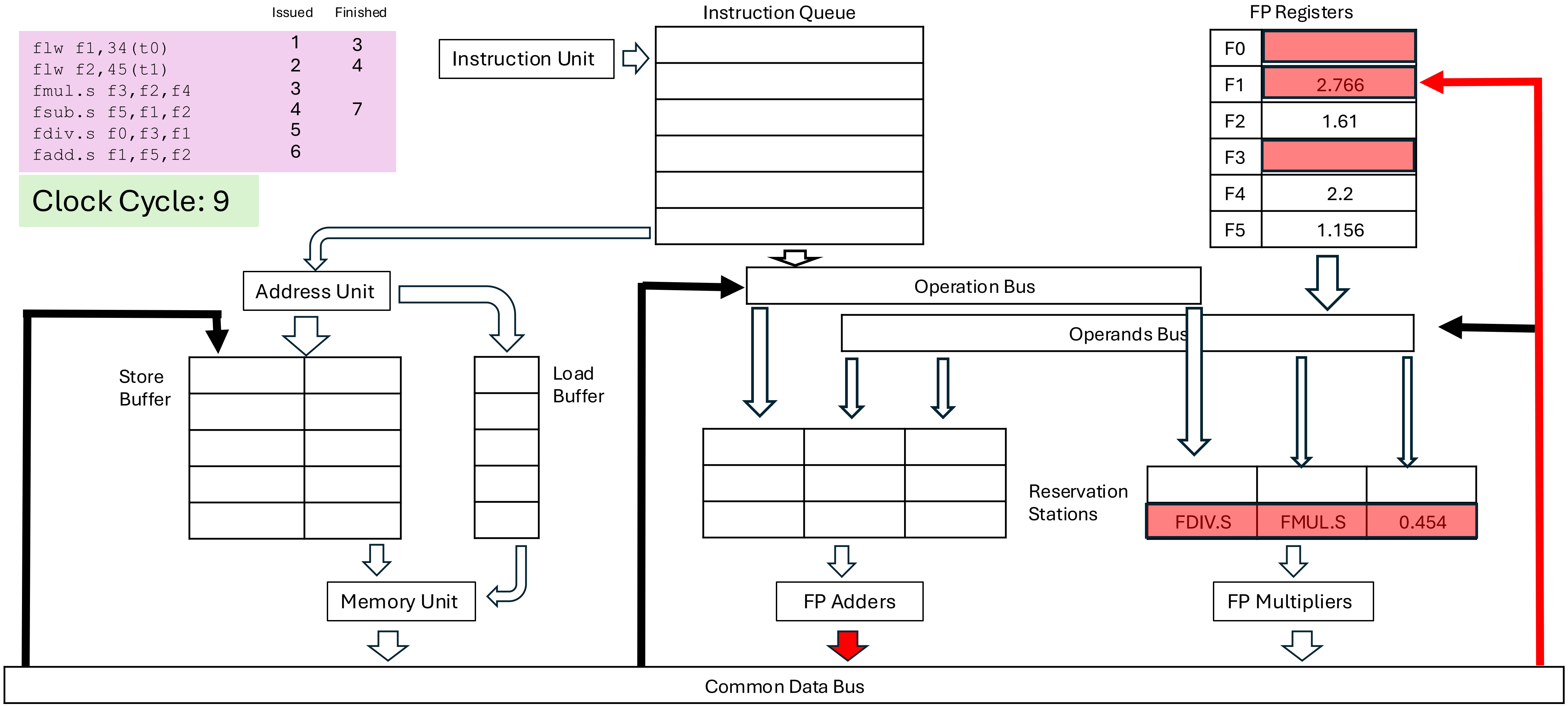
FP Registers

F0	🟥
F1	0.454
F2	1.61
F3	🟥
F4	2.2
F5	1.156

FADD.S	1.156	1.61

FDIV.S	FMUL.S	0.454

In clock cycle 8 the `fadd.s f1,f5,f2` is sent to the FP Adders. Register F5 is no longer busy waiting.



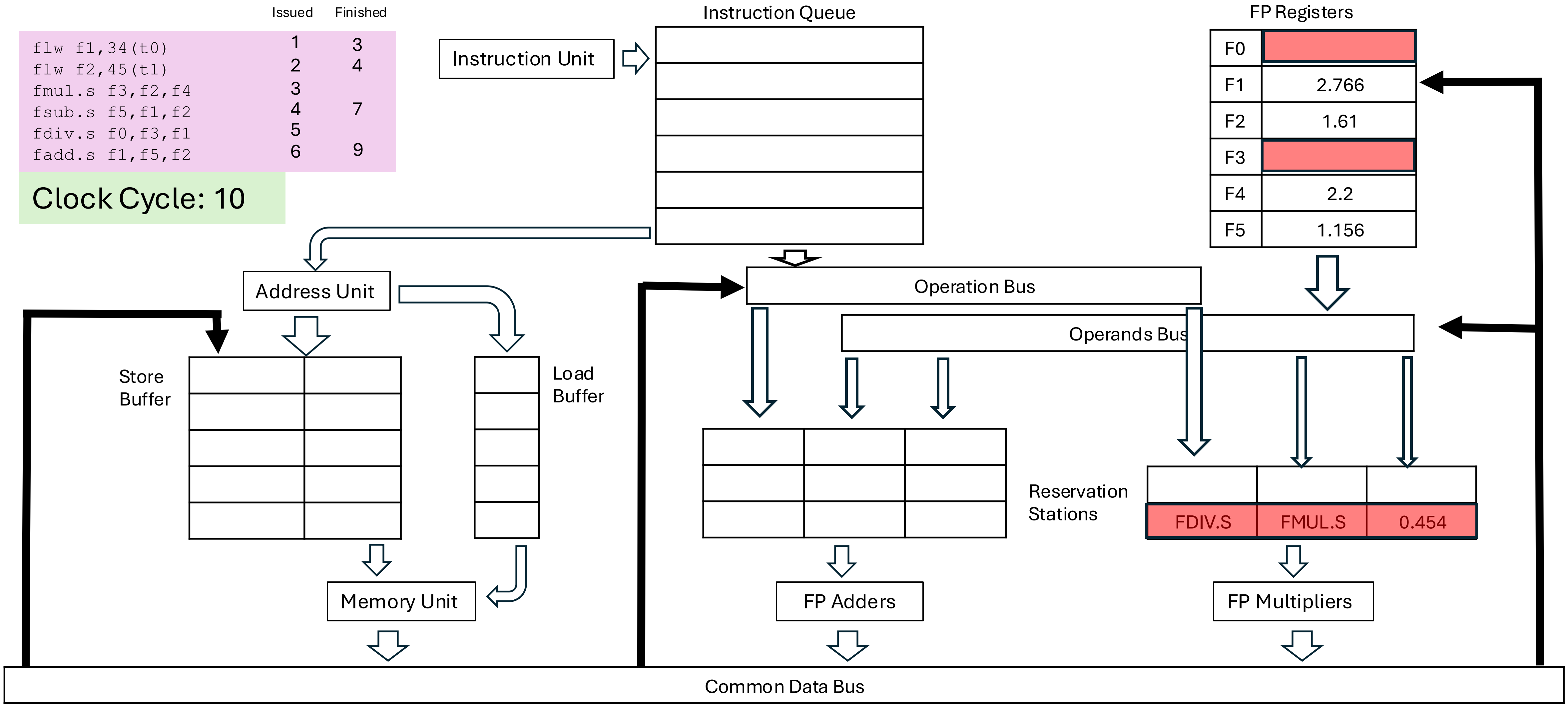
	Issued	Finished
`flw f1,34(t0)`	1	3
`flw f2,45(t1)`	2	4
`fmul.s f3,f2,f4`	3	
`fsub.s f5,f1,f2`	4	7
`fdiv.s f0,f3,f1`	5	
`fadd.s f1,f5,f2`	6	

Clock Cycle: 9

We are in clock cycle 9. The `fadd` operation was sent to the FP Adders in the earlier cycle, so its Reservation Station is free. Since it takes 2 cycles for anadd, Its results are written to the Common Data Bus and all subscribers, like, register F1.

	Issued	Finished
`flw f1,34(t0)`	1	3
`flw f2,45(t1)`	2	4
`fmul.s f3,f2,f4`	3	
`fsub.s f5,f1,f2`	4	7
`fdiv.s f0,f3,f1`	5	
`fadd.s f1,f5,f2`	6	9

Clock Cycle: 10



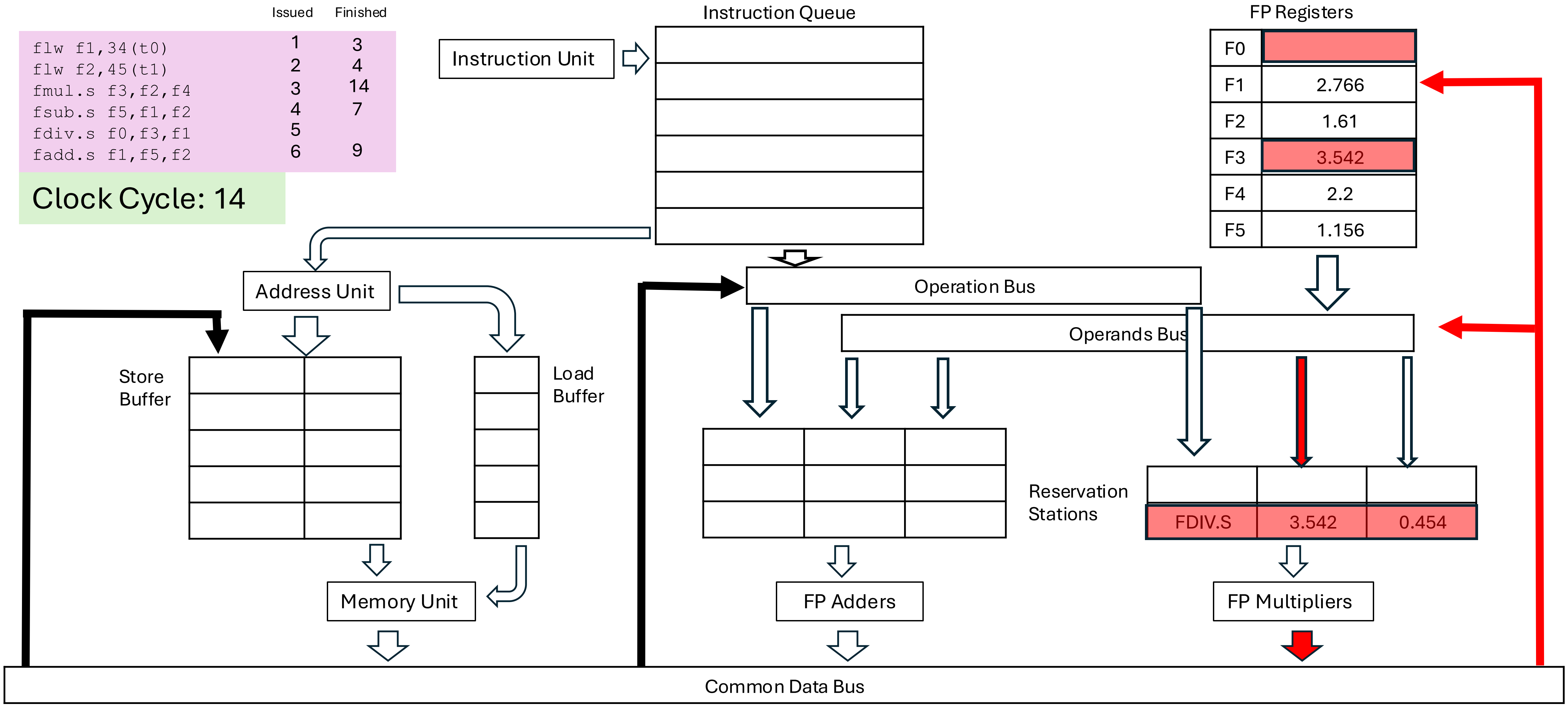
FP Registers

F0	🟥
F1	2.766
F2	1.61
F3	🟥
F4	2.2
F5	1.156

We are in clock cycle 10, register F1 is no longer busy.

	Issued	Finished
`flw f1,34(t0)`	1	3
`flw f2,45(t1)`	2	4
`fmul.s f3,f2,f4`	3	14
`fsub.s f5,f1,f2`	4	7
`fdiv.s f0,f3,f1`	5	
`fadd.s f1,f5,f2`	6	9

Clock Cycle: 14

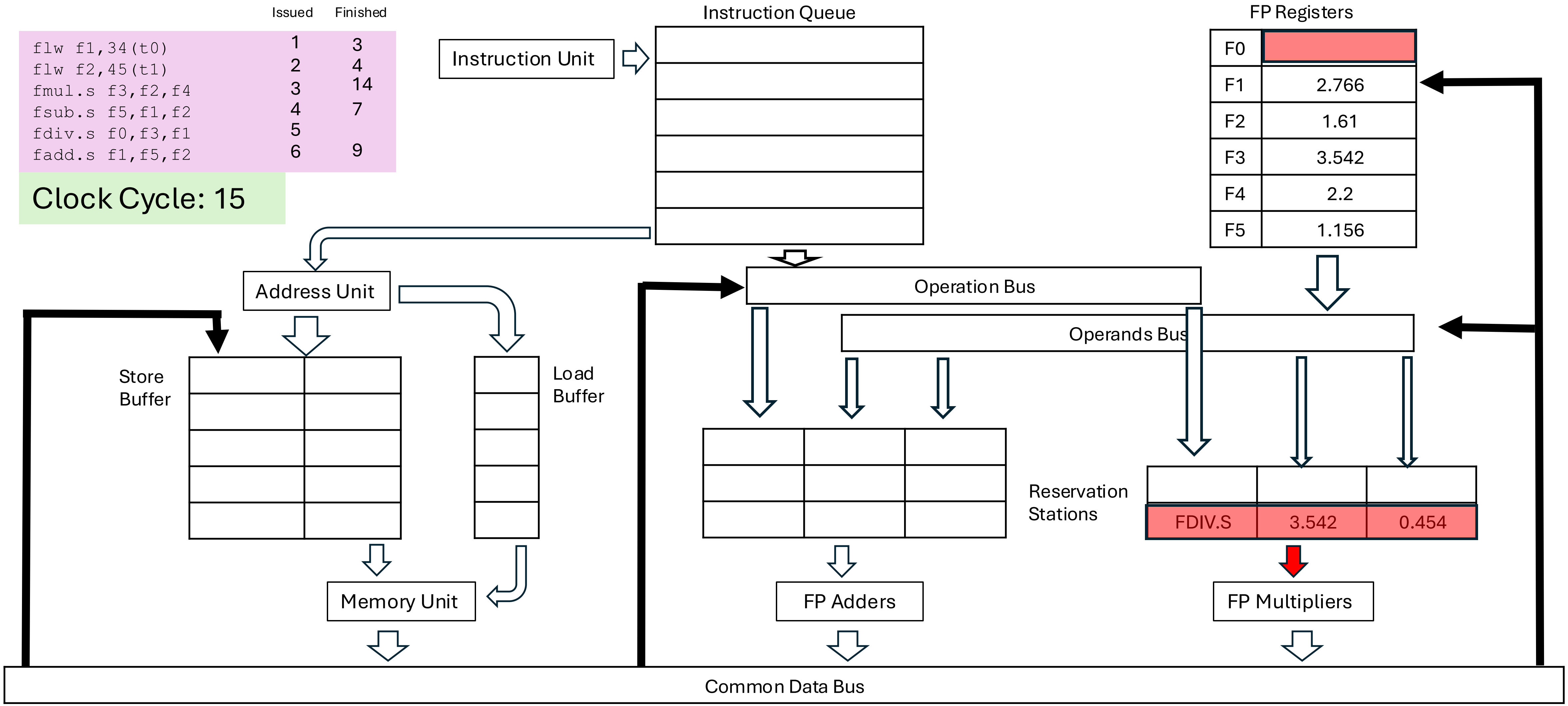


FP Registers

F0	
F1	2.766
F2	1.61
F3	3.542
F4	2.2
F5	1.156

FDIV.S	3.542	0.454

We are in clock cycle 14. The `fmul` operation has ended and its results are broadcast to the Common Data Bus and all its subscribers, that is, F3 register and the FDIV operation in the Reservation Station.



	Issued	Finished
`flw f1,34(t0)`	1	3
`flw f2,45(t1)`	2	4
`fmul.s f3,f2,f4`	3	14
`fsub.s f5,f1,f2`	4	7
`fdiv.s f0,f3,f1`	5	
`fadd.s f1,f5,f2`	6	9

Clock Cycle: 15

FP Registers

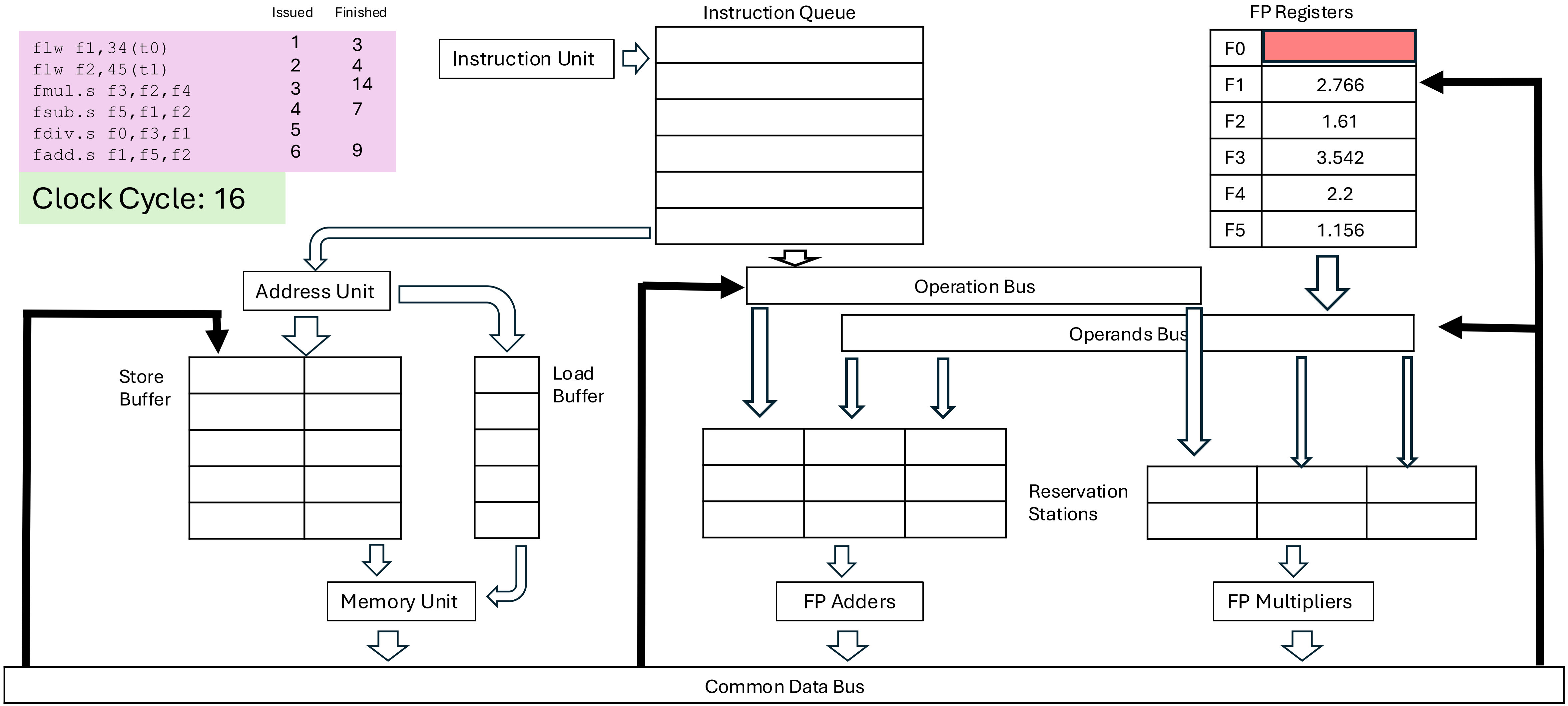
F0	🟥
F1	2.766
F2	1.61
F3	3.542
F4	2.2
F5	1.156

FDIV.S	3.542	0.454

We are in clock cycle 15. The `fdiv.s` operation has been sent to the FP Multipliers. Register F3 is no longer busy.

	Issued	Finished
`flw f1,34(t0)`	1	3
`flw f2,45(t1)`	2	4
`fmul.s f3,f2,f4`	3	14
`fsub.s f5,f1,f2`	4	7
`fdiv.s f0,f3,f1`	5	
`fadd.s f1,f5,f2`	6	9

Clock Cycle: 16

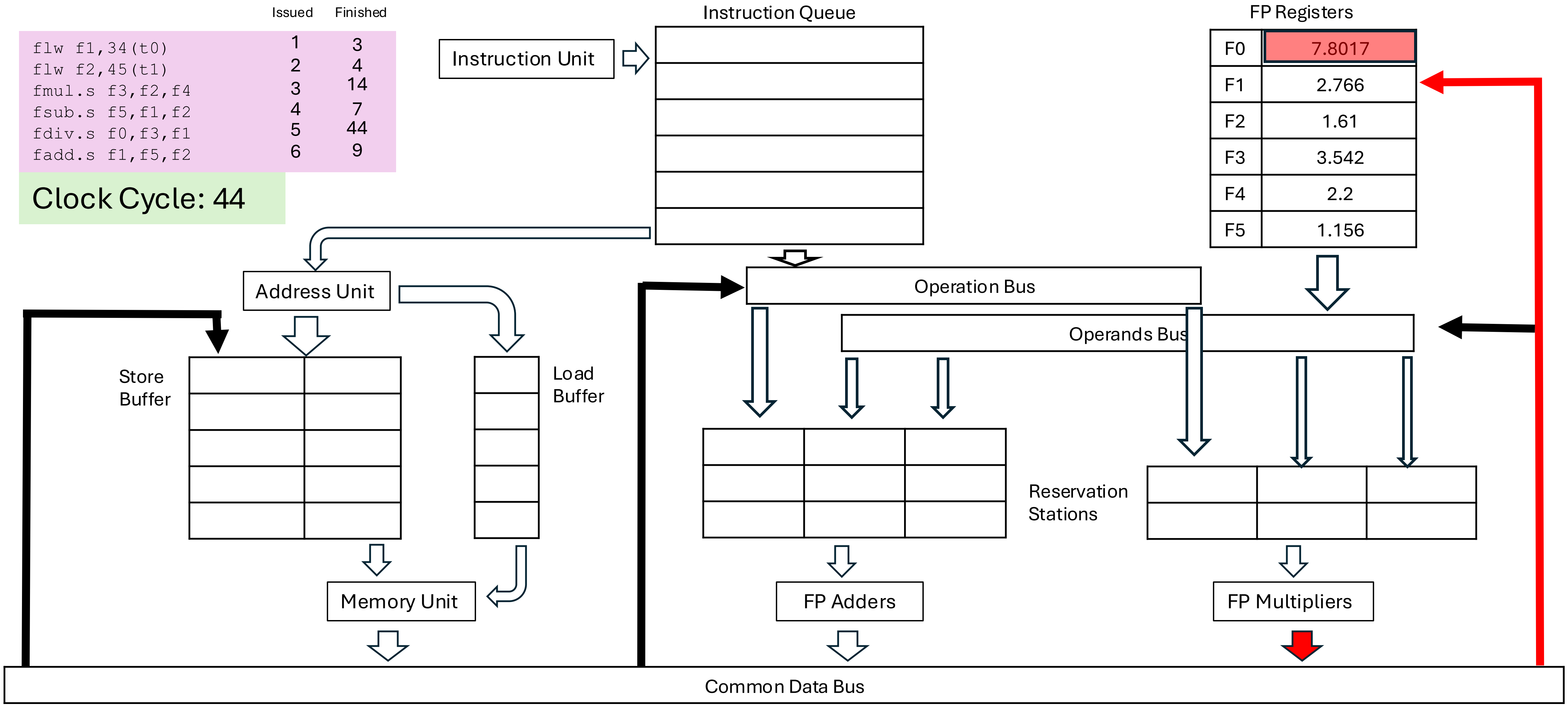


FP Registers	
F0	🟥
F1	2.766
F2	1.61
F3	3.542
F4	2.2
F5	1.156

Cycle 16. The Reservation Station containing the `fdiv.s` operation has been marked free.

	Issued	Finished
`flw f1,34(t0)`	1	3
`flw f2,45(t1)`	2	4
`fmul.s f3,f2,f4`	3	14
`fsub.s f5,f1,f2`	4	7
`fdiv.s f0,f3,f1`	5	44
`fadd.s f1,f5,f2`	6	9

Clock Cycle: 44



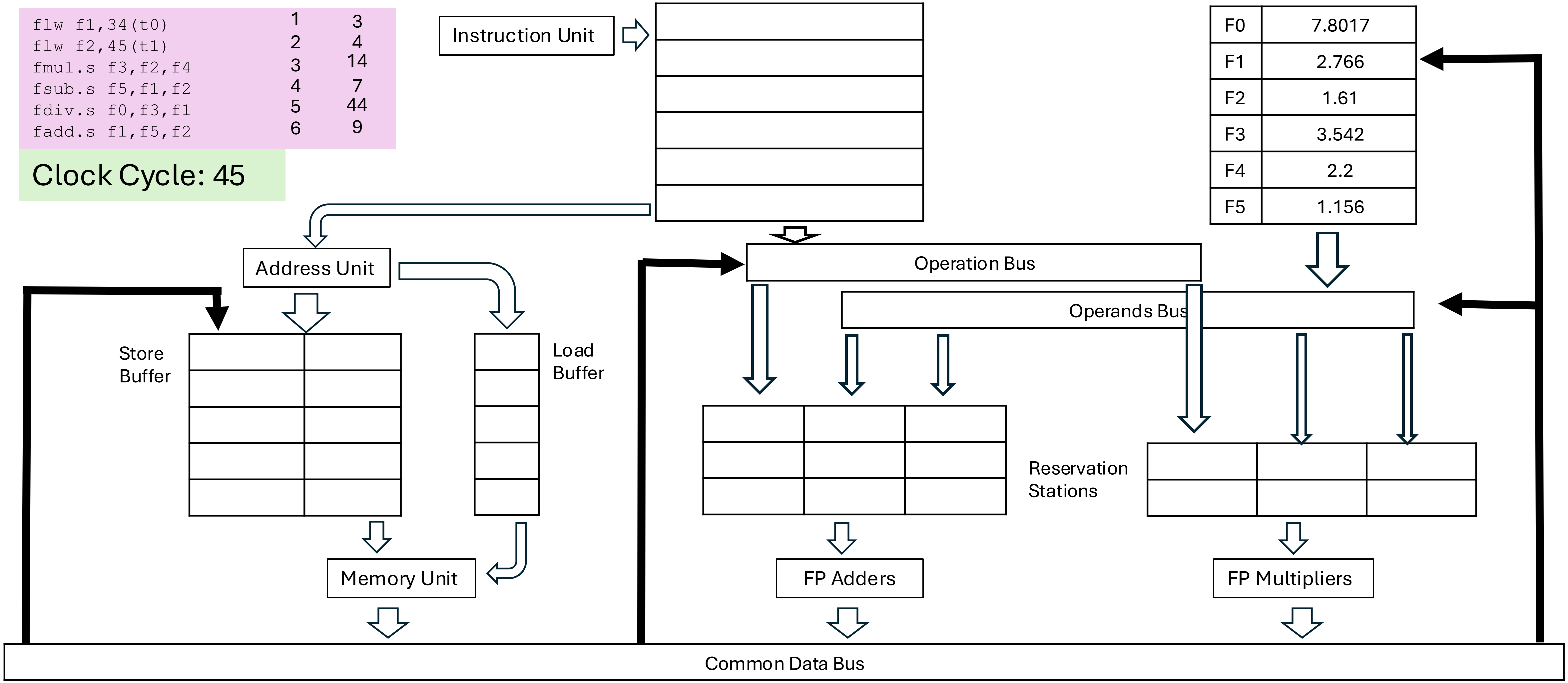
FP Registers

F0	7.8017
F1	2.766
F2	1.61
F3	3.542
F4	2.2
F5	1.156

We are in clock cycle 44. The `fdiv.s` operation has ended and its results are sent to the Common Data Bus and all its subscribers, which is only F0.

	Issued	Finished
`flw f1,34(t0)`	1	3
`flw f2,45(t1)`	2	4
`fmul.s f3,f2,f4`	3	14
`fsub.s f5,f1,f2`	4	7
`fdiv.s f0,f3,f1`	5	44
`fadd.s f1,f5,f2`	6	9

Clock Cycle: 45



FP Registers

F0	7.8017
F1	2.766
F2	1.61
F3	3.542
F4	2.2
F5	1.156

We are in clock cycle 45. F0 register is marked as free.