Computer Architectures
Exam of 09.07.2024 - part I

First name, Last name, ID.....

Question #2

Let's consider a superscalar MIPS64 architecture implementing dynamic scheduling, speculation, and multiple issues and composed of the following units:

- An issue unit able to process 2 instructions per clock period; in the case of a branch instruction, only one instruction is issued per clock period
- A commit unit able to process 2 instruction per clock period
- The following functional units (for each unit the number of clock periods to complete one instruction is reported):
  - 1 unit for memory access: 1 clock period
  - 1 unit for integer arithmetic instructions: 1 clock period
  - 1 unit for branch instructions: 1 clock period
  - 1 unit for FP multiplication (pipelined): 4 clock periods
  - 1 unit for FP division (unpipelined): 12 clock periods
  - 1 unit for other FP instructions (pipelined): 3 clock periods
  - 2 Common Data Bus.
- Let's also assume that:
  - Branch predictions are always correct
  - All memory accesses never trigger a cache miss.

You should use the following table to describe the behavior of the processor during the execution of the first 2 iterations of the loop, computing the total number of required clock cycles.
Tip: For the EXE stage, it is recommended to add a lowercase letter to help distinguish between the different functional units available.

Loop #	Instruction	ISSUE x2	EXE	MEM	CDB x2	COMMIT x2	Notes
1	l.d f1, v1(r1)	1	2m	3	4	5	
1	l.d f2, v2(r1)	1	3m	4	5	6	
1	mul.d f5, f1, f2	2	6x		10	11	Wait for f1(4), f2(5)
1	l.d f3, v3(r1)	2	4m	5	6	11	
1	l.d f4, v4(r1)	3	5m	6	7	12	
1	mul.d f6, f3, f4	3	8x		12	13	Wait for f3(6), f4(7), FP mul FU
1	sub.d f7, f5, f6	4	13a		16	17	Wait for f5(10), f6(12)
1	add.d f7, f7, f8	4	17a		20	21	Wait for f7(16), FP other FU
1	s.d f7, v5(r1)	5	6s			21	The address is calculated at once, then waits for f7(20)
1	daddui r1, r1, 8	5	6i		7	22	
1	daddi r2, r2, -1	6	7i		8	22	
1	bnez r2, loop	7	9b			23	Being critical, it was chosen to issue and commit the branches in separate CCs
2	l.d f1, v1(r1)	8	9m	10	11	24	
2	l.d f2, v2(r1)	8	10m	11	12	24	
2	mul.d f5, f1, f2	9	13x		17	25	Wait for f1(11), f2(12), FP mul FU
2	l.d f3, v3(r1)	9	11m	12	13	25	
2	l.d f4, v4(r1)	10	12m	13	14	26	
2	mul.d f6, f3, f4	10	15x		19	26	Wait for f3(13), f4(14), FP mul FU
2	sub.d f7, f5, f6	11	20a		23	27	Wait for f5(17), f6(19)
2	add.d f7, f7, f8	11	24a		27	28	Wait for f7(23), FP other FU
2	s.d f7, v5(r1)	12	13s			28	The address is calculated at once, then waits for f7(27)
2	daddui r1, r1, 8	12	13i		14	29	
2	daddi r2, r2, -1	13	14i		15	29	
2	bnez r2, loop	14	16b			30	Same as before

Small variations with respect to this solution could be accepted as correct in a case by case manner.