```
main:
daddui r1,r0,0          ;FDEMW 5
daddui r2,r0,100        ; FDEMW 1

loop:
l.d f1,v1(r1)           ;  FDEMW 1
l.d f2,v2(r1)           ;   FDEMW 1
l.d f3,v3(r1)           ;    FDEMW 1
l.d f4,v4(r1)           ;     FDEMW 1
div.d f6,f3,f4          ;      FDSddddddddMW 9
div.d f7,f1,f2          ;       FSDSSSSSSSddddddddMW 8
mul.d f5,f6,f7          ;         FSSSSSSSDSSSSSSSmmmmmmmmMW 8
s.d f5,v5(r1)           ;                 FSSSSSSSDESSSSSSsMW 1
daddui r1,r1,8          ;                         FDSSSSSSSEMW 1
daddi r2,r2,-1          ;                          FSSSSSSSDEMW 1
bnez r2,loop            ;                                  FSDEMW 2
halt                    ;                                    FXXXX 1
                        ;                                     FDEMW

                        ;6 + 35 * 100 = 3506
```

4 July 2011 -- Computer Architectures –- part 2/2

Name, Matricola .....

Question 2

Considering the same loop-based program, and assuming the following processor architecture for a superscalar MIPS64 processor implemented with multiple-issue and speculation:

- issue 2 instructions per clock cycle
- jump instructions require 1 issue
- handle 2 instructions commit per clock cycle
- timing facts for the following separate functional units:
  i. 1 Memory address 1 clock cycle
  ii. 1 Integer ALU 1 clock cycle
  iii. 1 Jump unit 1 clock cycle
  iv. 1 FP multiplier unit, which is pipelined: 8 stages
  v. 1 FP divider unit, which is not pipelined: 8 clock cycles
  vi. 1 FP Arithmetic unit, which is pipelined: 2 stages
- Branch prediction is always correct
- There are no cache misses
- There are 2 CDB (Common Data Bus).

o Complete the table reported below showing the processor behavior for the 2 initial iterations.

o

# iteration		Issue	EXE	MEM	CDB x2	COMMIT x2
1	l.d f1,v1(r1)	1	2M	3	4	5
1	l.d f2,v2(r1)	1	3M	4	5	6
1	l.d f3,v3(r1)	2	4M	5	6	7
1	l.d f4,v4(r1)	2	5M	6	7	8
1	div.d f6,f3,f4	3	14D		22	23
1	div.d f7,f1,f2	3	6D		14	23
1	mul.d f5,f6,f7	4	23X		31	32
1	s.d f5,v5(r1)	4	6M			32
1	daddui r1,r1,8	5	6I		7	33
1	daddi r2,r2,-1	5	7I		8	33
1	bnez r2,loop	6	9J			34
2	l.d f1,v1(r1)	7	8M	9	10	34
2	l.d f2,v2(r1)	7	9M	10	11	35
2	l.d f3,v3(r1)	8	10M	11	12	35
2	l.d f4,v4(r1)	8	11M	12	13	36
2	div.d f6,f3,f4	9	22D		30	36
2	div.d f7,f1,f2	9	30D		38	39
2	mul.d f5,f6,f7	10	39X		47	48
2	s.d f5,v5(r1)	10	12M			48
2	daddui r1,r1,8	11	12I		13	49
2	daddi r2,r2,-1	11	13I		14	49
2	bnez r2,loop	12	15J			50