**Question #1**

Let consider the Branch Prediction Mechanism based on the Branch Target Buffer (BTB).

You are requested to

1. Describe the architecture of a BTB
2. Describe in details the behavior of a BTB: when it is accessed, which input and output information are involved with each access
3. Assuming that the processor uses 32 bit addresses, each instruction is 4 byte wide, and the BTB is composed of 8 entries, clarify the content and size of the fields composing each BTB entry
4. Using the same assumptions of the previous point, identify the final content of the BTB if
   * The BTB is initially empty (i.e., full of 0s)
   * The following instructions are executed in sequence
     + l.d f1,v1(r1) located at the address 0x00A50050
     + bnez r2,l1 located at the address 0x00A50054; the branch is taken, and the branch target address is 0x00A60050
     + bez r4,l2 located at the address 0x00A60050; the branch is not taken.

**Question #2**

Let consider a superscalar MIPS64 architecture implementing dynamic scheduling, speculation and multiple issue 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 1 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): 6 clock periods
  + 1 unit for FP division (unpipelined): 8 clock periods
  + 1 unit for other FP instructions (pipelined): 2 clock periods
* 1 Common Data Bus.

Let 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 a cycle composed of the following instructions, computing the total number of required clock cycles. Registers f11 and f12 store two constants.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| # iteration |  | Issue | EXE | MEM | CDB | COMMIT |
| 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 | div.d f5, f3, f11 | 2 | 7d |  | 15 | 16 |
| 1 | sub.d f4, f1, f2 | 3 | 6f |  | 8 | 17 |
| 1 | add.d f6, f4, f5 | 3 | 16f |  | 18 | 19 |
| 1 | div.d f7,f6,f12 | 4 | 23 |  | 31 | 32 |
| 1 | s.d f7,v4(r1) | 4 | 5 |  | 32 | 33 |
| 1 | daddui r1,r1,8 | 5 | 6 |  | 7 | 34 |
| 1 | daddi r2,r2,-1 | 5 | 7 |  | 9 | 35 |
| 1 | bnez r2,loop | 6 | 10 |  |  | 36 |
| 2 | l.d f1,v1(r1) | 7 | 8 | 9 | 10 | 37 |
| 2 | l.d f2,v2(r1) | 7 | 9 | 10 | 11 | 38 |
| 2 | l.d f3,v3(r1) | 8 | 10 | 11 | 12 | 39 |
| 2 | div.d f5, f3, f11 | 8 | 15 |  | 23 | 40 |
| 2 | sub.d f4, f1, f2 | 9 | 12 |  | 14 | 41 |
| 2 | add.d f6, f4, f5 | 9 | 23 |  | 25 | 42 |
| 2 | div.d f7,f6,f12 | 10 | 31 |  | 39 | 43 |
| 2 | s.d f7,v4(r1) | 10 | 11 |  | 40 | 44 |
| 2 | daddui r1,r1,8 | 11 | 12 |  | 13 | 45 |
| 2 | daddi r2,r2,-1 | 11 | 13 |  | 16 | 46 |
| 2 | bnez r2,loop | 12 | 17 |  |  | 47 |