Considerare la seguente architectura MIPS64:

|  |  |  |
| --- | --- | --- |
| * + Integer ALU: 1 clock cycle   + Data memory: 1 clock cycle   + FP multiplier unit: pipelined 7 stages | * + FP arithmetic unit: pipelined 2 stages   + FP divider unit: not pipelined unit that requires 8 clock cycles   + branch delay slot: 1 clock cycle, and the branch delay slot disabled | * + forwarding enabled   + è possibile completare lo stage EXE di una istruzion in modo out-of-order. |

* Facendo riferimento al frammento di codice riportato, si mostrino le tempistiche relative all’esecuzione ciascuna istruzione e si calcoli il numero totale di clock cycles necessari per eseguire completamente il programma:

for (i = 0; i < 100; i++) {

v5[i] = ((v1[i]/v2[i]) + v3[i]);

v6[i] = ((v3[i]/v4[i]) + v1[i]\*v2[i]);

}

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| .data |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Clock  cycles |
| V1: .double “100 values” |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| V2: .double “100 values” |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| V3: .double “100 values”  …  V5: .double “100 zeros” |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| V4: .double “100 values” |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| V5: .double “100 values” |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| V6: .double “100 values” |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| .text |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| main: daddui r1,r0,0 | F | D | E | M | W |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 5 |
| daddui r2,r0,100 |  | F | D | E | M | W |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |
| loop: l.d f1,v1(r1) |  |  | F | D | E | M | W |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |
| l.d f2,v2(r1) |  |  |  | F | D | E | M | W |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |
| div.d f5,f1,f2 |  |  |  |  | F | D | s | : | : | : | : | : | : | : | : | M | W |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 9 |
| l.d f3,v3(r1) |  |  |  |  |  | F | s | D | E | M | W |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 0 |
| add.d f5, f5,f3 |  |  |  |  |  |  |  | F | D | s | s | s | s | s | s | a | a | M | W |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 2 |
| l.d f4, v4(r1) |  |  |  |  |  |  |  |  | F | s | s | s | s | s | s | D | E | s | M | W |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |
| div.d f6,f3,f4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | F | D | s | s | : | : | : | : | : | : | : | : | M | W |  |  |  |  |  |  |  |  |  |  | 9 |
| mul.d f1,f1,f2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | F | s | s | D | x | x | x | x | x | x | x | s | M | W |  |  |  |  |  |  |  |  |  | 1 |
| add.d f6,f6,f1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | F | D | s | s | s | s | s | s | s | + | + | M | W |  |  |  |  |  |  |  | 2 |
| s.d f5,v5(r1) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | F | s | s | s | s | s | s | s | D | E | s | M | W |  |  |  |  |  |  | 1 |
| s.d f6,v6(r1) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | F | D | E | s | M | W |  |  |  |  |  | 1 |
| daddui r1,r1,8 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | F | D | s | E | M | W |  |  |  |  | 1 |
| daddi r2,r2,-1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | F | s | D | E | M | W |  |  |  | 1 |
| bnez r2,loop |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | F | s | D | E | M | W |  | 2 |
| halt |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | F | s | D | E | M | W | 1 |
| Total | 6+33\*100 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 3306 |

Considerando il programma precedente, lo si ottimizzi in modo da eliminare per quanto possibile gli stalli del programma usando le tecniche note come rescheduling e register renaming. Si calcoli il tempo di esecuzione del nuovo programma nella stessa architettura evidenziando il miglioramento ottenuto.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| .data |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **Clock  cycles** |
| V1: .double “100 values” |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| V2: .double “100 values” |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| V3: .double “100 values”  …  V5: .double “100 zeros” |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| V4: .double “100 values” |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| V5: .double “100 values” |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| V6: .double “100 values” |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| .text |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| main: daddui r1,r0,0 | **F** | **D** | **E** | **M** | **W** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **5** |
| daddui r2,r0,100 |  | **F** | **D** | **E** | **M** | **W** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **1** |
| loop: l.d f1,v1(r1) |  |  | **F** | **D** | **E** | **M** | **W** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **1** |
| l.d f2,v2(r1) |  |  |  | **F** | **D** | **E** | **M** | **W** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **1** |
| l.d f3,v3(r1) |  |  |  |  | **F** | **D** | **E** | **M** | **W** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **0->1** |
| div.d f5,f1,f2 |  |  |  |  |  | **F** | **D** | **:** | **:** | **:** | **:** | **:** | **:** | **:** | **:** | **M** | **W** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **9->8** |
| l.d f4, v4(r1) |  |  |  |  |  |  | **F** | **D** | **E** | **M** | **W** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **1->0** |
| mul.d f1,f1,f2 |  |  |  |  |  |  |  | **F** | **D** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **M** | **W** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **1** |
| daddi r2,r2,-1 |  |  |  |  |  |  |  |  | **F** | **D** | **E** | **M** | **W** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **1->0** |
| div.d f6,f3,f4 |  |  |  |  |  |  |  |  |  | **F** | **D** | **s** | **s** | **s** | **s** | **:** | **:** | **:** | **:** | **:** | **:** | **:** | **:** | **M** | **W** |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **9->7** |
| add.d f5, f5,f3 |  |  |  |  |  |  |  |  |  |  | **F** | **s** | **s** | **s** | **s** | **D** | **+** | **+** | **M** | **W** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **2->0** |
| s.d f5,v5(r1) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **F** | **D** | **E** | **s** | **M** | **W** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **1->0** |
| add.d f6,f6,f1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **F** | **D** | **s** | **s** | **s** | **s** | **s** | **+** | **+** | **M** | **W** |  |  |  |  |  |  |  |  |  |  |  |  | **2** |
| s.d f6,v6(r1) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **F** | **s** | **s** | **s** | **s** | **s** | **D** | **E** | **s** | **M** | **W** |  |  |  |  |  |  |  |  |  |  |  | **1** |
| daddui r1,r1,8 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **F** | **D** | **s** | **E** | **M** | **W** |  |  |  |  |  |  |  |  |  |  | **1** |
| bnez r2,loop |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **F** | **s** | **D** | **E** | **M** | **W** |  |  |  |  |  |  |  |  |  | **2->1** |
| halt |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **F** | **D** | **E** | **M** | **W** |  |  |  |  |  |  |  |  | **1** |
| **Total** | **=6+25\*100=2506** | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | **2506** |

speedupenhanced= fractionenhanced=

speedupcomputation= speedupobservation=