Cody Martin

18-447 Homework 5

1.

a)

MOVI $R0, 0

MOVI $r1, 100

LEA $R2, A

LEA $R3, B

LEA $R4, C

LEA $R5, D

loop: LD $R6, $R3, $R0

LD $R7, $R4, $R0

MUL $R6, $R6, $R7

LD $R7, $R5, $R0

ADD $R6, $R6, $R7

RSHFA $R6, $R6, 1

ST $R6, $R2, $R0

ADD $R0, $R0, 1

ADD $R1, $R1, -1

BRnz loop

This takes 6 + 64\*100 = 6406

b)

LD VST, #1

LD VLN, #50

VLD V0, B

VLD V1, C

VLD V2, D

VLD V4, B + 50

VLD V5, C + 50

VLD V6, D + 50

Vmul V3, V0, V1

Vmul V7, V4, V5

Vadd V3, V3, V2

Vadd V7, V7, V6

Vrshfa V3, V3, 1

Vrshfa V0, V7, 1

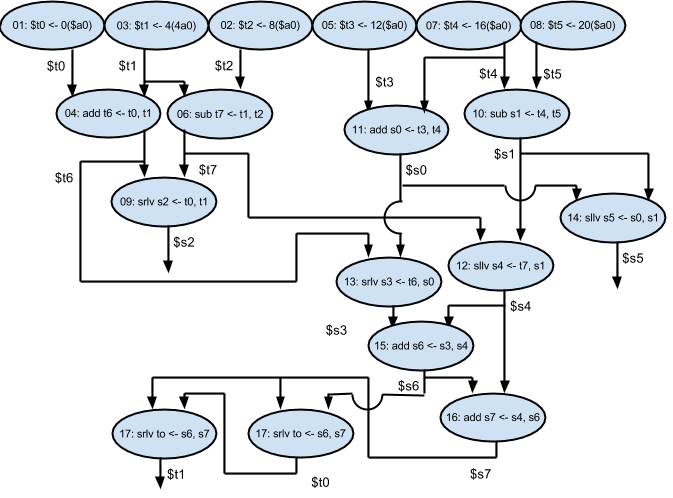
` VST V3, A

VST V7, A + 50

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Instruction | Operands | (i) no chaining, one memory port | (i) chaining, one memory port | (iii) chaining, 2 read ports, 1 write port |
| LD | VST, #1 | 1 | 1 | 1 |
| LD | VLN, #50 | 1 | 1 | 1 |
| VLD | V0, B | 60 | 60 | 60 |
| VLD | V1, C | 60 | 60 | 11 |
| VLD | V2, D | 60 | 60 | 49 |
| VLD | V4, B + 50 | 60 | 60 | 11 |
| VLD | V5, C + 50 | 60 | 60 | 49 |
| VLD | V6, D + 50 | 60 | 60 | 11 |
| Vmul | V3, V0, V1 | 55 | 6 | 6 |
| Vmul | V7, V4, V5 | 55 | 6 | 6 |
| Vadd | V3, V3, V2 | 53 | 47 | 47 |
| Vadd | V7, V7, V6 | 53 | 6 | 6 |
| Vrshfa | V3, V3, 1 | 50 | 44 | 44 |
| Vrshfa | V7, V7, 1 | 50 | 6 | 6 |
| VST | V3, A | 60 | 54 | 54 |
| VST | V7, A + 50 | 60 | 60 | 60 |
|  |  |  |  |  |
| (i) cycles: | 798 |  |  |  |
| (ii) cycles: | 591 |  |  |  |
| (iii) cycles: | 422 |  |  |  |

3.

a)



b) minimum N = 2

c)

|  |  |  |
| --- | --- | --- |
|  | MIPS Inst. # | MIPS Inst. # |
| VLIW instruction 1: | 1 | 2 |
| VLIW instruction 2: | 3 | 5 |
| VLIW instruction 3: | 7 | 8 |
| VLIW instruction 4: | 4 | 11 |
| VLIW instruction 5: | 6 | 10 |
| VLIW instruction 6: | 9 | 13 |
| VLIW instruction 7: | 12 | 14 |
| VLIW instruction 8: | 15 | 16 |
| VLIW instruction 9: | 17 | 18 |

d) N = 6

e)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | MIPS Inst. # | MIPS Inst. # | MIPS Inst. # | MIPS Inst. # | MIPS Inst. # | MIPS Inst. # |
| VLIW instruction 1: | 1 | 2 | 3 | 5 | 7 | 8 |
| VLIW instruction 2: | 6 | 10 |  |  |  |  |
| VLIW instruction 3: | 4 | 11 |  |  |  |  |
| VLIW instruction 4: | 9 | 13 |  |  |  |  |
| VLIW instruction 5: | 12 | 14 |  |  |  |  |
| VLIW instruction 6: | 15 | 16 |  |  |  |  |
| VLIW instruction 7: | 17 | 18 |  |  |  |  |

f)

Because of the parallel executing, the VLIW can execute multiple instructions at a time (the MIPS instructions loaded into each VLIW instruction). So if the execute stage of each instruction is greater than 1 cycle, the pipeline in the superscalar processor has to stall to wait for the execution unit to become available while the VLIW can start the execution whenever needed.

g)

If the program does not have any similar instructions that can be parallel executed, then the VLIW program will be the same as the MIPS program. It will take longer to execute a VLIW program with the same instructions as a MIPS program.

4. 

a) The TLB has 13, 4, 3

b)

|  |  |
| --- | --- |
| Frame Number | Frame Contents |
| 0 | Page 14 |
| 1 | Page 3 |
| 2 | Page 5 |
| 3 | Page 2 |
| 4 | Page6 |
| 5 | Page 4 |
| 6 | Page 15 |
| 7 | Page Table |

5.

a)

b)

c)

d)

6.

From sequence 1, we can tell that the block size is 8 bytes and that the cache size is 512B.

From sequence 2, we can tell that there is 4-way associativity.

From sequence 4, we can tell that the replacement protocol is LRU