Research & Reading 1

Comment on R2a

The paper compared performance of CISC and RISC on similar hardware and environment in details. As we all know, compared to CISC, RISC has many fewer cycles per instruction but more instructions per program, it is meaningful to figure out which is better as a whole.

In this paper, the author explained benchmark environment and conditions in detail, and conducted in-depth analysis of benchmark results, which greatly improved the credibility.

According to this paper, RISC had significantly higher architecturally-determined performance than the CISC on benchmarks. At the same time, different compiler performance, insufficient data and different operating systems may cause inaccurate results. Researchers are supposed to consider all these factors in the future.

Comment on R2b pages 1 - 35

The paper discussed how much ILP (Instruction-level parallelism) exists in typical programs and considered factors such as branch prediction, register renaming and alias analysis. The author conducted in-depth analysis of those factors.

Features from more important to less important

Branch Prediction, alias analysis, register renaming, jump prediction, simultaneous speculative execution across different paths.

Branch prediction with speculative execution was used to reduce stalling when control hazard occurs, which was the most important factor of ILP.

Comment on its weaknesses

The result in this paper was based on many optimistic assumptions like unlimited resources, no penalty for a missed prediction, all machines had same cycle time and all machines were built with comparable technology. Those assumptions may influence on results and should be cared by readers.

Research & Reading 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **SATA** | **SCSI** | **PCI-X** | **My Choice** |
| *Data Width* | 128 bits | 16 bits | 64 bits | SATA |
| *Clock Rate* | 125 MHz | 160 MHz | 133 MHz |  |
| *Bandwidth* | 16 Gbps | 2560 Mbps | 8528 Mbps |  |

Exercise 1

Answer

Total Cost to double the MTTF of the system = $7,500,000.

After double the MTTF, system catastrophic failure period increased from 10 days to 20 days.

That means the number of catastrophic failures within the same time will be halved.

So, whether it is a good business decision depends on the cost per catastrophic failure, only if the cost is over $7,500,000, then it will be a good business decision to double the MTTF.

Exercise 2

Answer

a)

b)

Processor B is faster than processor A.

Performance ratio (processor B / processor A) is 142.9%.Exercise 3

Answer

With full forwarding, only lw instructions with dependencies need a one cycle stall.

Execution Diagram

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| addi, $t6, $t6, 10 | IF | ID | EX | MEM | WB |  |  |  |  |  |  |  |
| sub $t5, $t6, $t4 |  | IF | ID | EX | MEM | WB |  |  |  |  |  |  |
| srl $t5, $t5, 2 |  |  | IF | ID | EX | MEM | WB |  |  |  |  |  |
| sw $t5, 20($t5) |  |  |  | IF | ID | EX | MEM | WB |  |  |  |  |
| lw $t2, 0($t6) |  |  |  |  | IF | ID | EX | MEM | WB |  |  |  |
| add $t7, $t2, $t3 |  |  |  |  |  | IF | ID | \*\*\* | EX | MEM | WB |  |
| beq $t5, $t7, End |  |  |  |  |  |  | IF | \*\*\* | ID | EX | MEM | WB |

Exercise 4

Answer

a) 4 cycles are lost to branch overhead per loop iteration.

b) 2 cycles are wasted on branch overhead.

c) No lost on correct predictions.

Exercise 5

Answer

a) Non-pipelined (single cycle) processor

b) Pipelined processor

c) Choose S3 stage

Case Study

Answer

Pipeline Execution Diagram

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| lw, r2, 0(r1), 10 | IF | ID | EX | MEM | WB |  |  |  |  |  |  |  |  |  |  |
| beq r2, r0, label2 |  | IF | ID | \*\*\* | EX | MEM | WB |  |  |  |  |  |  |  |  |
| sw r1, 0(r2) |  |  | IF | \*\*\* | ID | break |  |  |  |  |  |  |  |  |  |
| add r4, r5, r6 |  |  |  |  | IF | break |  |  |  |  |  |  |  |  |  |
| lw r3, 0(r2) |  |  |  |  |  | IF | ID | EX | MEM | WB |  |  |  |  |  |
| beq r3, r0, label1 |  |  |  |  |  |  | IF | ID | \*\*\* | EX | MEM | WB |  |  |  |
| beq r2, r0, label2 |  |  |  |  |  |  |  | IF | \*\*\* | ID | EX | MEM | WB |  |  |
| sw r1, 0(r2) |  |  |  |  |  |  |  |  |  | IF | ID | EX | MEM | WB |  |
| add r4, r5, r6 |  |  |  |  |  |  |  |  |  |  | IF | ID | EX | MEM | WB |