Review Problem 53

Absolute value, without using any branches.

In assembly, replace the value in X0 with its

\[
\text{CASE } x_0 \rightarrow x_1, x_0, \text{ or } 0
\]

\[
\text{SUBS } x_1, x_3; x_0 \text{ if } x_1 = -x_0
\]

\[
\text{SUB } x_1, x_0; x_0 \text{ if } x_1 = 0
\]

\[
\text{LSR } x_2, x_0; x_0 \text{ if } x_2 = 1
\]
Simple Branching

Schedule across branch
Not backwards compatible
Not every parallelism

Concerns

0.25
Max CPI

Very Long Instruction Word
<table>
<thead>
<tr>
<th>Branch</th>
<th>Load/Store</th>
<th>ALU2</th>
<th>ALU</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>door</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(6)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>door</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>door</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(1)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Instructions in parallel with a branch will execute. Schedule the code for a 4-way VLIW. Assume no delay slots, and all

VLIW Scheduling
Dynamically schedule multiple instructions based on hazard detection.
CPU? No delay slots, instructions in parallel with a branch still execute

Example Execution on Modern Processors